OP-19
AMO-3140C
20 June 2002
NAS8-39073

Chandra X-Ray Observatory

CXO Operations Database User's Guide

Post-Launch Baseline

Changes through OFLS Release 9.3

Chandra X-Ray Center
60 Garden Street
Cambridge, MA 02138

Chandra X-Ray Observatory
Operations Database User's Guide

OP-19 Revision C
20 June 2002

Prepared by: _____

Paul A. Levitt

Approved by:

_____

Roger Brissenden
Manager, Chandra X-Ray Center

_____

Dan Schwartz
Manager, Science Operations Team

_____

Chris Eagan
Manager, Operations Control Center

_____

Jeff Holmes
Manager, Ground System Engineering

_____

Leon McKendrick
Flight Operations Manager

Symbol	Date	Authorization	Revision/Change Description	Pages Affected
C	6/20/02		First revision under SAO Configuration Control. Numerous editorial changes, corrections. Incorporates HPRs and ECRs through OFLS Release 9.3, including: AX000712/SCN031 AX000730/SCN035 AX000732/SCN036 AX000746 OCCcm03405 OCCcm05104	ALL
B	7/16/99	G.J. Lee R.S. Watson, Jr PDMO Release	Rev B Incorporates the following MSFC CCB Approved PCNs: AX000584/SCN 013 (RECP R100) AX000600/SCN 014 (ECP E0172) AX000597/SCN 015 (ECP E01 73) AX000615/SCN 016 (RECP R105 AX000585/SCN 017 (ECP E0161) AX000590/SCN 018 (RECP R101) AX000630/SCN 019A (ECP AXGS-0017) AX000636/SCN 020 (RECP R108) AX000656/SCN 021 (RECP AXGS-R0082 & R110)AX000675/SCN 022 (ECP E01 83) AX000692/SCN 023 (ECP AXGS-0024) AX000677/SCN 024 AX000695/SCN 026 AX000681/SCN 025 (ECP 99-096) AX000699/SCN 028 (RECP R0093) AX000696/SCN 029 (RECP R1 15 & R0089) AX000717/SCN 030 AX000701/SCN 027 AX000726/SCN 032 AX000727/SCN 033 AX000728/SCN 034	ALL
A	3/5/98	C. Langton	Rev A incorporates the following MSFC	ALL

Symbol	Date	Authorization	Revision/Change Description	Pages Affected
		R. S. Watson, jr. PDMO Release (signatures on file)	CCB Approved PCNs: AX000267 AX000298 AX000305 AX000341 AX000435 AX000438 AX000441 AX000447 AX000464 AX000475 / SCN 001 AX000487 / SCN 002 AX000488 / SCN 003 AX000489 / SCN 004 AX000490 / SCN 005 AXOW507 / SCN 006 AXOW512 / SCN 007 AX000455 / SCN 008 AX000558 / SCN 009 AX000539 / SCN 010 AXOW536 / SCN 011 AX000515 / SCN 012 Changes incorporated per MSFC Approved (PCN AX000526) with changes per CCBD XC3-00-0123.	
–	1/6/97	Original CADM Release N. Vargas (signature on file)	Initial Release Incorporating baseline comments: GP54B-96-186 and GP548-96-207	ALL

Table of Contents

Table of Contents

Table of Contents

Table of Contents

Table of Contents

List of Tables

List of Tables

1. INTRODUCTION

1.1 PURPOSE

This is the User's Guide for the Chandra X-ray Observatory (CXO) Operations Database (ODB). It is to be used as a reference describing the content and format of the elements in the ODB, and provides background information including identification of the source documents for element definitions, identification of the organizations responsible for element definitions, format, and data, descriptions of the methods and media used for element transfer, and references to instructions for the maintenance, control, use and verification/validation of the ODB. This information is required to enable the Chandra Operations Control Center (OCC) / Engineering Support Center (ESC) to develop, maintain and operate the software systems used to process telemetry, generate commands, display data, and support CXO Operations personnel in carrying out Chandra Mission Operations.

1.2 SCOPE

This user's guide defines the content and format of each ODB element and provides additional descriptive information on the use of the element in support of CXO operations.. Format descriptions of some ODB elements are explicitly included in the User's Guide while others are provided indirectly by reference to other documents. Format descriptions are a detailed specification of element structure including element format (flat file - a single table in a relational database - or relational database), number of fields, field names, field sizes, field data types, field location in records or tables, and the element type, such as ASCII. The actual data in the fields is not provided.

ODB element background information is provided to facilitate ODB control and manipulation. When specified, the "value provider" field identifies the organization responsible for providing information. Element transfer media descriptions specify information required for the recipient of the element to retrieve the data from the delivery medium in its proper form for use in the OCC. Element identification data includes name, version number, creation date, and providing organization.

The ODB User's Guide is maintained at the Chandra Operations Control Center by the Systems Engineering Team, and is controlled by the Flight Director Board. It will be updated periodically to correctly reflect the current ground system software and operational procedures.

1.3 REFERENCE DOCUMENTS

The following documents form a part of this user's guide to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this user's guide, the contents of this user's guide prevails.

AMO-1001	AXAF Integrated Operations Schedule
AMO-1140/D22734/ TRW DR OP20	AXAF Operations Database Management Plan
AMO-2000	AXAF Operations Ground System Requirements
AMO-2050	OCC to ASC ICD
AMO-2080	ONLS to OFLS ICD
AMO-2130/IF1-60/ TRW DR CM07d	AXAF to OCC ICD
AMO-2140	AXAF Operations Control Center (OCC) to Deep Space Network (DSN) Interface Control Document (ICD)
AMO-2300	AXAF OFLS System Software Requirements
AMO-2310	AXAF OFLS Software Design Specification
AMO-3110/OP05	AXAF Constraints , Restriction, and Limitations
AMO-3130/TRW DR OP03	AXAF Systems and Operation Procedures
HOSC-EHS-065	Enhanced Huntsville Operations Support Center (HOSC) System (EHS) Concepts and Scenarios Document.
HOSC-SDD-044	Scripting Services Software Design Document
HOSC-SDD-048	Display Services Software Design Document
HOSC-SDD-049	Computations Services Software Design Document
HOSC-SRS-019	Software Requirements Specification for the Command Database Processing CSCI of the Enhanced HOSC System
HOSC-SRS-022	Software Requirements Specification for the Telemetry Database Processing CSCI of the Enhanced HOSC System
HOSC-SRS-069	Software Requirements Specification for the Off-Line Data Elements Management CSCI of the Enhanced HOSC System
HOSC-SYS-072	HOSC System Test and Validation Plan
MSFC-DOC-1949	Marshall Space Flight Center (MSFC) Huntsville Operations Support Center (HOSC) Telemetry and Command Database Definition

MSFC-PLAN-904	HOSC Functional Requirements and Implementation Plan
MSFC-RQMT-1440	Generic Requirements for the Enhanced HOSC System
MSFC-STD-1274	MSFC HOSC Telemetry Format Standard
TRW DR CM07F/ IF11-62	OCC to SMF ICD
TRW DR DM05	AXAF Software Requirements Specification, Volume 1A - On-Board Computer. Volume 1B - Control Processing Electronics
TRW DR SE17	AXAF Instrumentation Program and Command List (IP&CL)

1.4 ACRONYM LIST

AC	Aspect Camera
ACA	Aspect Camera Assembly
ACIS	AXAF CCD Imaging Spectrometer
AD&SC	Attitude Determination and Spacecraft Control
ADS	Aspect Determination System
AGASC	AXAF Guide and Aspect Star Catalog
ANSI	American National Standards Institute
ASC	AXAF Science Center
ASCDS	ASC Data System
ASCII	American Standard Code for Information Interchange
ATS	Absolute Time Sequence
AXAF	Advanced X-Ray Astrophysics Facility
BEP	Back-End Processor
CARD	Constraints and Restrictions Document
CCD	Charge Coupled Device
CCDM	Communications, Command and Data Management
CCP	Central Command Processor
CD	Compact Disk
CDB	Command Database
CDR	Critical Design Review
CM	Command Management
CMD	Command
CPE	Control Processing Electronics
CRC	Cyclic Redundancy Check
CSC	Computer Science Corporation (OFLS)
CSCI	Computer System Component Item
CSS	Coarse Sun Sensor
CTU	Command And Telemetry Unit
CXO	Chandra X-Ray Observatory
CXC	Chandra X-Ray Center

ACRONYM LIST

DBCG	Database Coordination Group
DBCR	Database Change Request
DEC	Declination
DEG	Degree
DOC	Document
DOT	Detailed Operations Timeline
DSN	Deep Space Network
ECI	Earth Centered Inertial
ECR	Engineering Change Request
EEPROM	Electrically Erasable Programmable Read Only Memory
EHS	Enhanced HOSC System
EIO	EPHIN Input/Output
EPHIN	Electron Proton Helium Instrument
EPS	Electrical Power System
ESA	Earth Sensor Assembly
ESC	Engineering Support Center
FA	Focused Assembly
FID	Fiducial
FITS	Flexible Image Transfer System
FMT	Format
FORTRAN	FORTRAN programming language
FOT	Flight Operations Team
FOV	Field Of View
FSS	Fine Sun Sensor
FSW	Flight Software
GMT	Greenwich Mean Time
GS	Ground Systems
GSC	Guide Star Catalog
GSTDN	Ground Space Tracking and Data Network
HDR	Header

ACRONYM LIST

HETG	High Energy Transmission Grating
HOSC	Huntsville Operations Support Center
HRC	High Resolution Camera
HRC-I	High Resolution Camera - Imaging
HRC-S	High Resolution Camera - Spectrometer??
HRMA	High Resolution Mirror Assembly
HW	Hardware
ICD	Interface Control Document
I-EPHIN	Integrated-EPHIN
IOE	Input-Output Electronics
IOS	Integrated Operations Schedule
IP&CL	Instrumental Programs And Command List
IRU	Inertial Reference Unit
ISS	Interface & Support Software
IU	Interface Unit
LOS	Loss of Signal
JPL	Jet Propulsion Laboratory
LETG	Low Energy Transmission Grating
LGA	Low Gain Antenna
MAX	Maximum
MDI	Mechanical Design Integration
MIN	Minimum
Mission Comp	Mission Computations Software
MOL	Mission Operations Laboratory (ONLS)
MPS	Mission Planning And Scheduling
MSFC	Marshall Space Flight Center
MSID	Measurement/Stimulus Identifier
MUPS	Momentum Unloading Propulsion System
NASCOM	NASA Communication
NAV	Navigation Subsystem
NRT	Near Real Time

ACRONYM LIST

NSSDC	National Space Science Data Center
OBC	On-Board Computer
OBS	Observation Statement
OCC	Operations Control Center
OCDB	Operational Command Database
ODB	Operations Database
ODE	Off-Line Data Element Management
OFF	On-board Flight Firmware
OFLS	Off-Line System
OFP	On-board Flight Program
ONLS	On-Line System
OPS	Operations
OR	Observation Request
PCAD	Pointing Control And Attitude Determination
PDR	Preliminary Design Review
PPM	Positions and Proper Motion
RA	Right Ascension
RAM	Random Access Memory
RCS	Reaction Control System
RCTU	Remote CTU
RDBMS	Relational Database Management System
ROM	Read Only Memory
RQMT	Requirement
RTS	Relative Time Sequence
SA	Solar Array
SC	Spacecraft
SCS	Stored Command Sequence
SDD	Software Design Document
SE	Systems Engineering
SGI	Silicon Graphics Inc.
SI	Science Instrument

ACRONYM LIST

SIM	Science Instrument Module
SLP	Solar, Lunar And Planetary
SMF	Software Maintenance Facility
SOE	Sequence of Event
SOT	Science Operations Team
SS&EA	Spacecraft Support & Engineering Analysis
SSA	Star Selection Algorithm
SSR	Solid State Recorder
S/W	Software
TBD	To Be Determined
TBR	To Be Resolved
TDB	Telemetry Database
TDM	Time Division Multiplexed
TLM	Telemetry
TOC	TYCHO Output Catalog
TSC	Translating Science Compartment
TRW	TRW Inc. (Spacecraft)
UDE	User Generated Data Element CSCI
UTC	Coordinated Universal Time
VCDU	Virtual Channel Data Unit
WCP	Workstation Command Processor

2. OPERATIONS DATABASE ELEMENTS

The ODB is comprised of three databases: the Command Database (CDB), the Telemetry Database (TDB), and the Off-Line Data Element (ODE) database. These databases are used as a data source in performing command processing, telemetry processing, mission planning and scheduling (MPS), command management (CM), attitude determination and sensor calibration (AD&SC), spacecraft support and engineering analysis (SS&EA), science instrument (SI) calibration, and AXAF data display and monitoring. Each of these databases is described in the corresponding Software Requirements Specification of the Enhanced HOSC System (HOSC-SRS-019, 022, 069).

Each of the 3 databases within the ODB consists of one or more data elements. Table 2-1 contains the current list of the ODB data elements and responsibilities. The ODB data elements are numbered and listed in alphabetical order on the element names. Each element is accessed by the Flight Operations Team (FOT) and/or Science Operations Team (SOT) using the tools defined in the ROUTINELY ACCESSED BY (S/W) column. The organization that provides the data element format is specified under FMT SOURCE (ORG.) column. The actual data content is submitted to the OCC by either FOT/SOT or automatically by an OCC software tool. This information is specified in column DATA ORIGINATOR.

Each ODB element is delivered to MSFC at different phases of the program to support Ground System Releases and End-to-End Tests. Detail delivery dates for the data elements are specified in the AXAF Integrated Operations Schedule (IOS) (AMO-1001).

Table 2-1 ODB Elements and Responsibilities

TYP	*NO*	*ODB ELEMENT*	*FMT SOURCE (ORG.)*	*DATA ORIGINATOR*	*ROUTINELY ACCESSED BY (S/W)*
ODE	1	NOT USED			
ODE	2	CHARACTERISTICS	CSC	FOT, SOT, CSC	OFLS (ALL)
CDB	3	COMMAND DEFINITION TABLES	MOL	FOT	ONLS (DBCR, CMD UPDATE APP.), OFLS (CM)
ODE	4	COMMAND DEFINITION TABLES, MULTI-PART	CSC	FOT	OFLS (CM)
ODE	5	COMMAND LOAD	CSC	OFLS (CM)	ONLS, OFLS (CM)
ODE	6	COMMAND LOAD IMAGE	CSC	OFLS (CM)	ONLS, OFLS (CM)
ODE	7	COMMAND SEQUENCE DEFINITIONS	CSC	FOT, SOT	OFLS (CM)
ODE	8	CONFIGURATION REFERENCE	CSC	FOT	OFLS (CM)
ODE	9	CONFIGURATION SNAPSHOT	CSC	OFLS (CM)	ONLS (MISSION COMP), OFLS (CM)
ODE	10	CONSTRAINTS	CSC	FOT, SOT, CSC	OFLS (ALL)

Table 2-1 ODB Elements and Responsibilities

TYP	*NO*	*ODB ELEMENT*	*FMT SOURCE (ORG.)*	*DATA ORIGINATOR*	*ROUTINELY ACCESSED BY (S/W)*
ODE	11	DSN APPROVED SCHEDULES	JPL	JPL	OFLS (MPS, CM)
ODE	12	DSN SCHEDULE REQUESTS	JPL	OFLS (MPS)	OFLS (MPS)
ODE	13	ENGINEERING REQUEST	CSC	FOT, SOT	OFLS (MPS)
ODE	14	EPHEMERIS, DEFINITIVE	CSC	OFLS (ISS)	OFLS (ALL)
ODE	15	EPHEMERIS, PREDICTIVE	CSC	OFLS (ISS)	OFLS (ALL)
ODE	16	MCILWAIN PARAMETERS	CSC	OFLS (ISS)	ASC TOOL
ODE	17	MEMORY IMAGE, AC	TRW	SMF	ONLS (MISSION COMP)
ODE	18	MEMORY IMAGE, CPE	TRW	SMF	ONLS (MISSION COMP)
ODE	19	MEMORY IMAGE, CTU EEPROM	TRW	SMF	ONLS (MISSION COMP)
ODE	20	MEMORY IMAGE, I-EPHIN	TRW	SMF	ONLS (MISSION COMP)
ODE	21	MEMORY IMAGE, IU EEPROM	TRW	SMF	ONLS (MISSION COMP)
ODE	22	MEMORY IMAGE, OBC	TRW	SMF	ONLS (MISSION COMP)
ODE	23	MEMORY IMAGE, SIM	TRW	SMF	ONLS (MISSION COMP)
ODE	24	OBSERVATION REQUEST	CSC	SOT	OFLS (MPS)
ODE	25	ORBIT EVENTS, DEFINITIVE	CSC	OFLS (ISS)	OFLS (ALL except AD&SC)
ODE	26	ORBIT EVENTS, PREDICTIVE	CSC	OFLS (ISS)	OFLS (ALL except AD&SC)
ODE	27	RADIATION ZONE DEFINITIONS	CSC (NSSDC)	FOT (NSSDC)	OFLS (ISS)
ODE	28	RELATIVE TIME SEQUENCE	CSC	FOT, SOT	OFLS (CM)
ODE	29	SCHEDULED OR/ER DATA	CSC	OFLS (MPS)	ASC TOOL
ODE	30	SENSOR CALIBRATION DATA	CSC	OFLS (AD&SC)	OFLS (AD&SC)
ODE	31	SOFTWARE UPDATES, AC	TRW	SMF	OFLS (CM)
ODE	32	SOFTWARE UPDATES, ACIS	ASC	ASC	OFLS (CM)
ODE	33	SOFTWARE UPDATES, CPE	TRW	SMF	OFLS (CM)
ODE	34	SOFTWARE UPDATES, CTU EEPROM	TRW	SMF	OFLS (CM)
ODE	35	SOFTWARE UPDATES, I-EPHIN	TRW	SMF	OFLS (CM)
ODE	36	SOFTWARE UPDATES, IU EEPROM	TRW	SMF	OFLS (CM)
ODE	37	SOFTWARE UPDATES, OBC	TRW	SMF	OFLS (CM)

Table 2-1 ODB Elements and Responsibilities

TYP	*NO*	*ODB ELEMENT*	*FMT SOURCE (ORG.)*	*DATA ORIGINATOR*	*ROUTINELY ACCESSED BY (S/W)*
ODE	38	SOFTWARE UPDATES, SIM	TRW	SMF	OFLS (CM)
ODE	39	SOLAR, LUNAR, PLANETARY DATA	CSC	JPL	OFLS (ALL except CM)
ODE	40	SPACECRAFT CLOCK CORRELATION	CSC	OFLS (SS&EA)	OFLS (ALL)
ODE	41	STAR CATALOG	ASC, CSC	SOT	OFLS (MPS, AD&SC, SS&EA)
ODE	42	STATE VECTORS FROM DSN NAV	CSC	JPL	OFLS (ISS)
ODE	43	TABLES, ACIS	ASC	SOT	OFLS (CM)
ODE	44	Deleted			
ODE	45	Deleted			
TDB	46	TELEMETRY DEFINITION TABLES	MOL	FOT	ONLS (DBCR, CMD UPDATE APP.)

This page intentionally left blank

3. ODB ELEMENT DESCRIPTIVE INFORMATION

This section contains the format specifications for every data element and is organized with each section number corresponding to the ODB data element number in Table 2-1. Every element description section is further divided into three sub-sections: Header Information, Content, and Format Description. The Header Information lists the element type, responsible organization for defining the format, responsible organization for providing the data content, organization(s) requiring routine access to the element, element format structure such as ASCII flat file, Relational Database Table, etc., and delivery media. The Content sub-section provides the element definition, and the purpose for the element. The Format Description sub-section describes the element format stored in the ODB. In most cases, the element will be delivered in the same format as stored in the ODB, except those stated in the delivery media in the Header sub-section.

Each element is defined and accessed by different organizations through various software tools, hence, the format description for each element varies according to the format provider. The format description contains one or more of the following information:

- file organization
- estimated file size
- detailed description of the record layout

The record layouts are described in terms of a record identifier/name, the record format, the maximum length, and a record description table detailing every field within the record. The record description table contains the field name, a description of the field, the field format, the dimension of the data in the field, and the total field length.

Data element file names can take the form name.ext, where the ext field may be optional. Data element file names can be no longer than 25 characters in length excluding any extension. Extensions can be no longer than 4 characters including the period (.). Data element specific naming conventions are included in the description of that data element.

Table 3-1 lists the nomenclature used in this section to describe an element format.

Table 3-1 Format Description Nomenclature

Term	*Definition*
Absolute time	An ASCII string specifying GMT or an orbit event time. GMT times are specified in HOSC standard format as yyyy:ddd:hh:mm:ss.sss. Absolute times specifying GMT must include the year. Milliseconds may be omitted and a value of 0 will be assumed. (see HOSC GMT Time or Extended HOSC GMT Time for subfield definitions). Orbit event times are specified in the format (ORB, orbit_number, orbit_event, +/-relative time). (see Orbit Event Time for detail definition).
Array	A field defined in terms of its indices (e.g., Index: 1 = solar array, 2 = join point vector) and the dimension (e.g., 2x3). Indices are not specifically defined for vectors (dimension = 3) or matrices (dimension = 3x3).

Table 3-1 Format Description Nomenclature

Term	*Definition*
	Arrays are stored in column-row format, with the first index increasing most rapidly. Thus a 2x3 array is specified as ARRAY(1,1), ARRAY(2,1), ARRAY(1,2), ARRAY(2,2), ARRAY(3,1), ARRAY(3,2).
ASCII	Character string field format.
ASCII file	A file in ANSI standard ASCII format where individual values in an ASCII file record can be integer, real, logical, or ASCII format. Unless otherwise noted in the file format description for an individual data element, each record in an ASCII sequential file is assumed to be an individual line in the file (terminated with a line feed character).
ASCII time	A field format of ASCII character strings in either absolute times or relative times format (refer to Absolute time and Relative time).
Binary file	A file in binary code (0's and 1's) where individual fields in a binary file record are in SGI UNIX (internal) format, and can be integer (4 bytes), real (8 bytes), logical, or ASCII format.
Dimension	Index structure of a field. See also array, single dimension.
Direct access file	A file in which the records can be read by record number. Records in a direct access file can be read sequentially (see sequential file) or can be read by specifying a record number within the file.
Extended GMT Format	An ASCII string format of yyyy:ddd:hh:mm:ss.sss with subfield definitions as follows: Y - Represents four digits for a year. If y is not defined, then the time is assumed to be relative. D - Represents three digits for a day, and cannot exceed 366 days and is measured from Greenwich midnight, December 31, preceding the year specified. H - Represents two digits for hours. Hours are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time. M - Represents two digits for minutes. Minutes are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time. S - Represents two digits for seconds and three digits for milliseconds after the decimal. Seconds are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time.
Field Format	Format of a field within a record. See also integer, logical, real, long, ASCII, ASCII time.
Fixed record	A record format that always has the same length as specified for the record length.
GMT Format	An ASCII string in the format yyyy:ddd:hh:mm:ss where the subfield definitions are as follows: Y - Represents four digits for a year. If y is not defined, then the time is assumed to be relative.

Table 3-1 Format Description Nomenclature

Term	*Definition*
	D - Represents three digits for a day, and cannot exceed 366 days and is measured from Greenwich midnight, December 31, preceding the year specified. H - Represents two digits for hours. Hours are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time. M - Represents two digits for minutes. Minutes are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time. S - Represents two digits for seconds. Seconds are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time.
Indexed file	A file in which the records can be accessed by 1 or more keys. Records in an indexed file can be read sequentially (see sequential file) or can be read by supplying a key identifier, a value relationship (e.g., Equal to or greater than), and an expected value.
Integer / Int.	A field format of ANSI standard signed integers without decimal points. Unless otherwise specified in the length column, integer fields are 4 bytes for non-namelist files.
Length	The length specifies either the number of characters that can be specified for a field in an ASCII file or the number of bytes in a field for a binary or namelist file.
Logical	A field format with 1 or 4 byte that can have a value of true or false only; where true has the value of 1 and false of 0.
Long	A field format with variable length character strings containing up to 2 gigabytes, or 2^{31} - 1. Long field type is used for storing data images in hexadecimal codes where the field needs to be large enough to hold the entire image.
Orbit Event Time	Orbit event time specification is provided as an option on absolute time specifications. The processing resolves the statement time by referencing the associated time for the specified event in the orbit event file and then applies an optional positive or negative relative time. Orbit event time specifications take the form of TIME=(ORB,orbit_number,orbit_event,+/relative_time).
Real	A field format of ANSI standard FORTRAN real numbers (F , D, and E format) with double precision SGI IRIX floating point numbers. Unless otherwise specified in length column, real fields are 8 bytes for non-namelist files.
Record Format	Record length type, either fixed or variable. See also variable record and fixed record.
Relative Time	An ASCII string specifying a delta GMT in the format ddd:hh:mm:ss.sss (see Extended GMT Time for subfield definitions). Relative times do not include the year, but can include any other time subfield. A relative time must include at a minimum the hours, minutes, and seconds fields. The day field and/or the millisecond field may be omitted and a value of 0 will be assumed.

Table 3-1 Format Description Nomenclature

Term	*Definition*
Sequential file	A file in which the records must be read in the file order.
Single dimension	A field defined without indices and with no dimension specified in the dimension column.
Variable record	A record format that varies in length up to a maximum value as specified for the record length.

The ODB elements in the following sections are ordered alphabetically by the element names. The format descriptions for all the data elements are provided jointly by the following organization(s):

AXAF Science Center (ASC)

Jet Propulsion Laboratory (JPL)

Computer Science Corporation (CSC)

Mission Operations Laboratory (MOL)

TRW Mission Operations (TRW)

3.1 ACA CONFIGURATIONS

3.1.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: ASC

Data Provided By: ASC

Data Routinely Accessed By: OFLS, ASC

Format Structure: ASCII, Sequential

Delivery Media Electronic Transfer

3.1.2 CONTENT

This element contains entries which specify sets of ACA operating parameters for science observations and ACA calibrations. Each entry is indexed by a mnemonic which is referred to in the ACA_MODE parameter of an ER or OR. The file is flat ASCII, space-delimited, and has one line per entry.

3.1.3 FORMAT DESCRIPTION

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_aca_mnemonic	Name by which ER and OR refers to this entry	Char		8

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_aca_image_size	Size of ACA tracked image regions (4, 6, or 8 = 4x4, 6x6 mousebitten, or 8x8 pixels, respectively)	Integer		2
odb_aca_over_int_time	Override the default integration time which the ACA chooses	Logical		1
odb_aca_int_time	Use this value of integration time if override_aca_time is true (seconds)	Real		8
odb_aca_search_crit	In ACA search command, use nearest (0) or brightest (1) candidate image	Integer		1
odb_aca_convert_track	In ACA monitor command, specify that monitored images will be converted to track if bright enough	Logical		1
odb_aca_cal_type	ACA calibration type: dark current (0) or responsivity (1)	Integer		1
odb_aca_cal_col_sign	ACA calibration column numbers: positive (0) or negative (1) column numbers	Integer		1
odb_aca_cal_start_row	ACA calibration starting row number	Integer		2
odb_aca_cal_num_rows	ACA calibration number or rows read out in each integration	Integer		2

3.2 CHARACTERISTICS

3.2.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: TRW, ASC, CSC

Data Routinely Accessed By: OFLS

Format Structure: ASCII, Sequential

Delivery Media: Electronic Transfer

3.2.2 CONTENT

The AXAF characteristics file contains parameters used by the OFLS to define the spacecraft, spacecraft subsystems, and the basic mission including: the aspect camera (AC), the FSS, the CSSs, the ESA, the SIs, the LGA, the SSRs, communications, and the OBC.

Although this element will be maintained by the FOT, initial values for the data parameters contained in this element will be provided jointly by TRW (Spacecraft developer), ASC (Science Instrument Team), and CSC (OFLS developer).

3.2.3 FORMAT DESCRIPTION

The characteristic parameters are grouped into related records (as shown in the next section). Each record is formatted as an ANSI standard FORTRAN namelist:

$*namelist*

variable = value {[,value]} {!comment text}

{variable = value {[,value]} } {!comment text}

$*END*

Where

namelist is the name of the namelist (ex. record identifier column listed in the next section)

variable is the parameter to be included in the namelist (ex. Name column listed in the next section). The order in which the variable names appear is not important, nor do all the variables in the namelist have to appear.

value is the specified input value for the variable. If the variable is an array or a vector, the values will be listed in the order of the specified index (ex. Described in the dimension and description columns in next section)

$END is the terminating statement for each namelist.

comment text is the comments and must precede with an exclamation point !. A comment can be inserted as a stand-alone record or at the end of a parameter statement. Any text after the exclamation sign (!) will be treated as a comment and ignored by the interpreting program.

The record length is limited to 80 characters, parameters cannot be broken over individual records (records must end at a delimiter, either comma or space, between parameters), and there must be a space in column 1 of all records.

3.2.3.1 Data Records

This section describes the Characteristics data parameters requested by the OFLS (element format provider). The information described below were reviewed by TRW (element data provider). Due to discrepancies between algorithms adapted by OFLS and those used by Spacecraft software, initial values will be provided by TRW and CSC as specified in the initial value provider column in the following table. A cross-reference of the parameter to the OFLS Software Design Specification (AMO-2310) document section is also provided for additional reference.

Each record is formatted as an ANSI standard FORTRAN namelist.

FILE SIZE (ESTIMATED): 3,000 (Bytes)

FILE ORGANIZATION: FORTRAN Namelist

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_AC_Characteristics		**Variable**		**80**			
1	odb_ac_fov_y_min	Minimum aspect camera assembly (ACA) field of view (FOV) extent angle (size) in the negative y-angle measurement dimension (rotation about the ACA z-axis in the ACA frame) (degrees)	Real		8	TRW ACA	4.3.1.2.3.1, 4.3.2.3.6.1 (Determine Coarse and Fine Attitude, Simulate Sensor Telemetry)	$\alpha_{\min}$
2	odb_ac_fov_y_max	Maximum ACA FOV extent angle (size) in the positive y-angle measurement dimension (rotation about the ACA z-axis in the ACA frame) (degrees)	Real		8	TRW ACA	4.3.1.2.3.1, 4.3.2.3.6.1 (Determine Coarse and Fine Attitude, Simulate Sensor Telemetry)	$\alpha_{\max}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
3	odb_ac_fov_z_min	Minimum ACA FOV extent angle (size) in the negative z-angle measurement dimension (rotation about the ACA y-axis in the ACA frame) (degrees)	Real		8	TRW ACA	4.3.1.2.3.1, 4.3.2.3.6.1 (Determine Coarse and Fine Attitude, Simulate Sensor Telemetry)	$\beta_{\min}$
4	odb_ac_fov_z_max	Maximum ACA FOV extent angle (size) in the positive z-angle measurement dimension (rotation about the ACA y-axis in the ACA frame) (degrees)	Real		8	TRW ACA	4.3.1.2.3.1, 4.3.2.3.6.1 (Determine Coarse and Fine Attitude, Simulate Sensor Telemetry)	$\beta_{\max}$
5	odb_ac_search_rad	Defines the star field search radius for the star selection algorithm (degrees)	Real		8	ASC	4.5.4.19 (Obtain Stars from AXAF Guide and Acquisition Star Catalog)	ρ_{SSR}
6	odb_ac_min_pix_row	Defines the minimum pixel row number in the ACA FOV	Integer		4	ASC	4.5.4.19 (Obtain Data from AXAF Guide and Acquisition Star Catalog)	$\zeta^{ode}_{\min}$
7	odb_ac_max_pix_row	Defines the maximum pixel row number in the ACA FOV	Integer		4	ASC	4.5.4.19 (Obtain Data from AXAF Guide and Acquisition Star Catalog)	$\zeta^{ode}_{\max}$
8	odb_ac_min_pix_col	Defines the minimum pixel column number in the ACA FOV	Integer		4	ASC	4.5.4.19 (Obtain Data from AXAF Guide and Acquisition Star Catalog)	$\eta^{ode}_{\min}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
9	odb_ac_max_pix_col	Defines the maximum pixel column number in the ACA FOV	Integer		4	ASC	4.5.4.19 (Obtain Data from AXAF Guide and Acquisition Star Catalog)	$\eta_{\max}^{ode}$
10	odb_pixel_size	Defines the size of a pixel (arcseconds)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
11	odb_ac_fid_lt_mag	Fiducial light brightness expressed in ACA instrument magnitude for each fiducial light and each possible brightness setting (magnitude)	Real	14x256	28672	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
12	odb_ac_fid_lt_mag_err	Fiducial light instrumental magnitude error (index: fid light; magnitude)	Real	14	112	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
13	odb_ac_fid_nom_lev	Defines the fid light nominal brightness setting and the number of settings that this may be adjusted (index: 1=nominal setting, maximum increase in setting, maximum decrease in setting; 2 = FID light)	Integer	3x14	168	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
14	odb_rreg_width	Defines the width, dimension along the ACA y-coordinate, of the readout register (degrees; index: quadrant, 1 = +y, +z quadrant, 2 = +y, -z quadrant, 1 = -y, +z quadrant	Real	4	32	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	η_{rr}
15	odb_ac_mag_conv_flag	Indicates whether to treat star magnitudes from the AXAF star catalog as instrument magnitude or convert them using color intensity star data records in the AXAF star catalog. (1 = Convert, 0 = use directly)	Integer		4	ASC	4.3.1.2.3.1 (Determine Coarse and Fine Attitude)	N/A1
16	odb_ac_mag_00	ACA magnitude 0th order offset (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b
17	odb_ac_mag_01	ACA magnitude scale factor offset (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
18	odb_ac_mag_01_c1	ACA magnitude 1st order coefficient of color 1 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b
19	odb_ac_mag_01_c2	ACA magnitude 1st order coefficient of color 2 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b
20	odb_ac_mag_02_c1	ACA magnitude 2nd order coefficient of color 1 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b
21	odb_ac_mag_02_c2	ACA magnitude 2nd order coefficient of color 2 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.n (Determine Coarse and Fine Attitude, Select Acquisition Stars, Guide Stars and Fiducial Lights)	A_x^b
22	odb_ac_mag_03_c1	ACA magnitude 3rd order coefficient of color 1 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	a_x^b

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
23	odb_ac_mag_03_c2	ACA magnitude 3rd order coefficient of color 2 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	a_x^b
24	odb_ac_mag_04_c1	ACA magnitude 4th order coefficient of color 1 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	a_x^b
25	odb_ac_mag_04_c2	ACA magnitude 4th order coefficient of color 2 (indexed on magnitude band 1-50)	Real	50	400	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	a_x^b
26	odb_ac_mag_bands	Defines visual magnitude limits of the 10 magnitude bands used in calculating instrumental magnitude from visual magnitude. Values must be in ascending order. (index: magnitude band)	Real	11	88	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
27	odb_planet_mag	Defines the instrumental magnitude of the planets (index: Mercury, Venus, Mars, Jupiter, Saturn, Uranus, Neptune, Pluto)	Real	8	64	ASC	4.3.1.2.3.1, 4.1.3.12.3.2 (Determine Coarse and Fine Attitude, Calculate Instrumental Magnitude)	N/A
28	odb_angle_to_pix	Calibration parameters for computation of pixel coordinates of a star given the star angles in ACA coordinates during FOM processing (arcseconds^{-1})	Real			TRW ACA	4.1.3.12.3.6.5 Determine Roll Angles for Bad Pixel Region Interference4.1.3.12.n	$\Lambda_{Y,n}$, $\Lambda_{Z,n}$
29	odb_ac_nom_align	Defines the rotational transformation from the nominal ACA frame to the AXAF body frame	Real	3x3	72	TRW PCAD	4.3.1.1.7, 4.3.2.3.6.1 (Process Aspect Camera Assembly Centroids, Simulate Aspect Camera Assembly)	M_{mA}
30	odb_ac_misalign	Defines the rotational transformation from the misaligned ACA frame to nominal ACA frame	Real	3x3	72	TRW PCAD	4.3.1.1.7, 4.3.2.3.6.1 (Process Aspect Camera Assembly Centroids, Simulate Aspect Camera Assembly)	$M_{AA'}$
31	odb_ac_misalign_uncert	ACA to AXAF rotational transformation matrix uncertainty expressed as standard deviation on a per axis basis	Real	3	24	TRW PCAD	4.3.1.2.3.1(Generate Sub-catalog of Candidate Reference Stars)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
31A	odb_aca_trans_matrix	Defines the translation and rotation transformation between the ACA y-z coordinate system and linearized CCD pixel coordinates (index: 1= rotation matrix, 2 = ACA CCD in quadrant order)	Real	9x4	288	FOT	4.1.3.12.3.1 Obtain Stars from Star Catalog, 4.1.3.12.3.6.7.1 Calculate Roll Dependent Figure of Merit	M_{CA}
31B	odb_trans_polynomial	Coefficients defining the polynomial transformation between CCD pixels and y-z coordinates (index: 1= polynomial coefficient, 2 = y/z coordinate(1 = y-coordinate, 2 = z-coordinate), 3 = ACA CCD in quadrant order	Real	19x2x4	2116	BALL	4.1.3.12.2 Eliminate Fiducial Lights within Bad Pixel Regions	A_{ny}, A_{nz}
31C	odb_aca_temp	Defines the expected ACA temperature (degrees Celsius)	Real		8	FOT	4.1.3.12.2 Eliminate Fiducial Lights within Bad Pixel Regions,	T
32	odb_ac_bad_pixels	Bad pixel map expressed as the set of upper left corner pixel coordinates and lower right corner pixel coordinates of the bad zones (index: 1 = pixel coordinate (1 = y - pixel minimum, 2 = y – pixel maximum, 3 = z – pixel minimum, 4 = z – pixel maximum), 2 = bad zone)	Integer	4x100	1600	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
33	odb_ac_n_bad_pixels	Defines the number of entries in the bad pixel map, odb_ac_bad_pixels	Integer		4	TRW ACA	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
34	odb_ac_ang_meas_error	Standard deviation of the measurement noise for ACA centroid measurements (index: 1 = y-direction, 2 = z-direction, degrees)	Real		8	ASC	4.3.2.3.6 (Simulate Aspect Camera Assembly)	σ_a
35	odb_ac_mag_meas_error	Standard deviation of the measurement noise for ACA image magnitudes (degrees)	Real		8	ASC	4.3.2.3.6 (Simulate Aspect Camera Assembly)	σ_m
36	odb_ac_ang_meas_mean	Mean of the measurement noise for ACA centroid measurements (index: 1 = y-direction, 2 = z-direction, degrees)	Real		8	ASC	4.3.2.3.6 (Simulate Aspect Camera Assembly)	μ_a
37	odb_ac_mag_meas_mean	Mean of the measurement noise for ACA image magnitudes (index: 1 = y-direction, 2 = z-direction, degrees)	Real		8	ASC	4.3.2.3.6 (Simulate Aspect Camera Assembly)	μ_m
38	odb_min_sep_ac	Defines the minimum separation distance on the AC focal plan for any two directed search locations, excluding fid lights and planets (pixels)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	τ_{exel}
39	odb_min_sep_fid	Defines the minimum separation distance on the AC focal plane for a star directed search location and a fid light directed search location (pixels); index = fid light	Real	14	8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	τ_{exel}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
40	odb_min_sep_pln	Defines the minimum separation for a star or FID light directed search location and a planet (pixels)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	τ_{exel}
41	odb_min_rad_mag_g	Defines the difference in magnitude that a star or a planet must be brighter than a candidate guide star to be considered a radial spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}
42	odb_min_rad_mag_a	Defines the difference in magnitude that a star or a planet must be brighter than a candidate acquisition star to be considered a radial spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}
43	odb_min_rad_mag_f	Defines the difference in magnitude that a star or a planet must be brighter than a fid light to be considered a radial spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}
44	odb_min_rreg_mag_g	Defines the difference in magnitude that a star image on the readout register must be brighter than a candidate guide star to be considered a readout register spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
45	odb_min_rreg_mag_a	Defines the difference in magnitude that a star image on the readout register must be brighter than a candidate acquisition star to be considered a readout spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}
46	odb_min_rreg_mag_f	Defines the difference in magnitude that a star image on the readout register must be brighter than a fid light to be considered a readout register spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	Δmag_{rad}
47	odb_min_col_sep	Defines the minimum separation for column spoiler check (pixels)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	τ_{cel}
48	odb_min_col_mag_g	Defines the difference in magnitude that a star must be brighter than a candidate guide star to be considered a column spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	Δmag_{rad}
49	odb_min_col_mag_a	Defines the difference in magnitude that a star or planet must be brighter than a candidate acquisition star to be considered a column spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
50	odb_min_col_mag_f	Defines the difference in magnitude that a star or planet must be brighter than a FID light to be considered a column spoiler	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
51	odb_sbox_g	Defines the half-width of the guide star search box (pixels)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
52	odb_sbox_f	Defines the half-width of the fid light search box (pixels)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
53	odb_hires	Defines the method of onboard computation of the image search box half width from the search region dimension, D = 0, half-width = 20 + 40*D = 1, half-width = 20+ 5*D	Real		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
54	odb_brt_mag_r	Defines the limiting bright star magnitude (in instrumental magnitude) during responsivity calibration	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
55	odb_brt_mag_ac	Defines the limiting bright star magnitude (in instrumental magnitude) for acquisition stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
56	odb_fnt_mag_ac	Defines the limiting faint star magnitude (in instrumental magnitude) for acquisition stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
57	odb_brt_mag_g	Defines the limiting bright star magnitude (in instrumental magnitude) for guide stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
58	odb_fnt_mag_g	Defines the limiting faint star magnitude (in instrumental magnitude) for guide stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
59	odb_fnt_vmag_g	Defines the limiting faint visual magnitude allowed for guide stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
60	odb_fnt_vmag_a	Defines the limiting faint visual magnitude allowed for guide stars	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
61	odb_co_mag_10	Defines the ACA count accumulation rate for a star of instrumental magnitude ten (counts/sec)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
62	deleted						4.1.3.12.n	N/A[1]
63	odb_count_threshold	Defines the flux level above which positional uncertainty does not improve (counts)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
64	odb_sigma_p1	Defines the first coefficient of positional uncertainty	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
65	odb_sigma_p2	Defines the second coefficient of positional uncertainty	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
66	deleted						4.1.3.12.n	N/A[1]
67	odb_par_signal_loss	Defines the signal loss of parallel (Z) readout transfers	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars , Guide Stars and Fiducial Lights))	N/A[1]
68	odb_ser_signal_loss	Defines the signal loss of serial (Y) readout transfers	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars , Guide Stars and Fiducial Lights))	N/A[1]
69	odb_aca_err_coef	Defines the coefficient for the additive ACA error array	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
70	deleted						4.1.3.12.n	N/A[1]
71	odb_aca_t_err_coef	Defines the coefficient for the temperature dependent additive ACA error array	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
72	deleted						4.1.3.12.n	N/A[1]
73	odb_aca_a_err_coef	Defines the coefficient for the Sun angle dependent additive ACA error array	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
74	deleted						4.1.3.12.n	N/A[1]
75	odb_max_fom	Defines the maximum acceptable figure of merit for accepting guide stars (arcseconds^2)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
76	odb_cmd_int	Defines commanded integration time for responsive calibration (relative time)	ASCII		12	ASC	4.1.3.12.n (Select Acquisition Stars , Guide Stars and Fiducial Lights))	N/A[1]
77	odb_lsm_dc_time	Defines the commanded integration times allowed during dark current calibration for the corresponding limiting star magnitude (relative time)	ASCII	3	36	ASC	4.1.3.12.n (Select Acquisition Stars and Guide Stars)	N/A[1]
78	odb_fid_int_ac	Defines the specified intensity of the fiducial lights as a function of integration time (specified in most cases via threshold)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
79	odb_lasm_as	Defines the limiting apparent star motion during automatic search (arcseconds/second)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]
80	odb_lasm_st	Defines the limiting apparent star motion during directed search and track (arcseconds/second)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
81	odb_ac_warmup	Defines the minimum AC warm-up time (relative time)	ASCII		12	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]
82	odb_ac_ccd_id	Identifies the CCD array in use: 1 = primary, 2 = backup	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]
83	odb_dcc_time	Defines the commanded integration time for dark current calibration (relative time)	ASCII		12	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]
84	odb_min_acq	Defines the minimum required number of commanded acquisition stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights))	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
85	odb_max_acq	Defines the desired maximum number of commanded acquisition stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
86	odb_min_guide	Defines the minimum required number of commanded guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
87	odb_max_guide	Defines the desired maximum number of commanded guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
88	odb_min_c1_g	Defines the minimum allowable value for the first quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
89	odb_max_c1_g	Defines the maximum allowable value for the first quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
90	odb_min_c2_g	Defines the minimum allowable value for the second quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
91	odb_max_c2_g	Defines the maximum allowable value for the second quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
92	odb_min_c3_g	Defines the minimum allowable value for the third quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
93	odb_max_c3_g	Defines the maximum allowable value for the third quality code for guide stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
94	odb_min_c_a	Defines the minimum allowable value for any quality code for acquisition stars	Integer		4	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
95	odb_roll_lever_arm	Defines the scaling factor for the roll error when calculating the figure of merit (arcminutes)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A
95A	odb_acq_dur	Defines the maximum duration of an AC acquisition, including acquiring separate acquisition and guide stars (seconds)	Real		8	TRW	N/A	
95B	odb_mag_aca_nsig	Defines the number of standard deviations of random and internal star magnitude error tolerated when acquiring stars onboard	Real		8	ASC	4.1.3.12.3.2 (Calculate Instrumental Magnitude)	η

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
95C	odb_mag_aca_err_ran	Defines the estimated additional random error in guide or acquisition star magnitudes	Real		8	ASC	4.1.3.12.3.2 (Calculate Instrumental Magnitude)	δ_{ran}
95D	odb_mag_aca_err_sys	Defines the estimated systematic error in guide or acquisition star magnitude	Real		8	ASC	4.1.3.12.3.2 (Calculate Instrumental Magnitude)	δ_{sys}
	ODE_CSS_Characteristics		**Variable**		**80**			
96	odb_css_max_current	Coarse Sun sensor (CSS) current output corresponding to the Sun along the detector boresight (milliamperes)	Real		8	TRW PCAD	4.3.1.1.4, 4.3.2.3.3 (Process Coarse Sun Sensor Data, Simulate Coarse Sun Sensor)	I_{max}, I_{peak}
97	odb_css_boresight_angle	Angle between the solar array y-axis to the CSS boresight in the solar array plane (degrees)	Real		8	TRW PCAD	4.3.1.1.4 (Process Coarse Sun Sensor Data)	ζ
98	odb_css_i_noise_mean	CSS output current bias (dark current) (milliamperes)	Real		8	TRW PCAD	4.3.2.3.3 (Simulate Coarse Sun Sensor)	μ_{CSS}
99	odb_css_i_noise_std_dev	CSS output current noise standard deviation (milliamperes)	Real		8	TRW PCAD	4.3.2.3.3 (Simulate Coarse Sun Sensor)	σ_{CSS}
	ODE_ESA_characteristics		**Variable**		**80**			
100	odb_esa_nom_align	Defines the rotational transformation from the nominal Earth sensor assembly (ESA) frame to the AXAF body frame for each ESA; index: 3 = ESA	Real	3x3x2	144	TRW PCAD	4.3.1.1.6, 4.3.2.3.5 (Process Earth Sensor Assembly Data, Simulate Earth Sensor Assembly)	M_{mE}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
101	odb_esa_misalign	Defines the rotational transformation from the misaligned ESA frame to nominal ESA frame for each ESA; index: 3=ESA	Real	3x3x2	144	TRW PCAD	4.3.1.1.6, 4.3.2.3.5 (Process Earth Sensor Assembly Data, Simulate Earth Sensor Assembly)	$M_{EE'}$
102	odb_esa_cone_angle	Angle between the center of the ESA instantaneous FOV and the ESA scan axis for each ESA (degrees)	Real	2	16	TRW PCAD	4.3.1.1.6, 4.3.2.3.5 (Process Earth Sensor Assembly Data, Simulate Earth Sensor Assembly)	γ
103	odb_esa_phase_meas_mean	ESA Earth scan crossing angle measurement bias for each ESA(degrees)	Real	2	16	TRW PCAD	4.3.2.3.5 (Simulate Earth Sensor Assembly)	$\mu_{\alpha ESA}$
104	odb_esa_phase_meas_std_dev	ESA Earth scan crossing angle measurement noise standard deviation for each ESA (degrees)	Real	2	16	TRW PCAD	4.3.2.3.5 (Simulate Earth Sensor Assembly)	$\sigma_{\alpha ESA}$
105	odb_esa_chord_meas_mean	ESA Earth chord angle measurement bias for each ESA(degrees)	Real	2	16	TRW PCAD	4.3.2.3.5 (Simulate Earth Sensor Assembly)	$\mu_{\Omega ESA}$
106	odb_esa_chord_meas_std_dev	ESA Earth chord angle measurement noise standard deviation for each ESA (degrees)	Real	2	16	TRW PCAD	4.3.2.3.5 (Simulate Earth Sensor Assembly)	$\sigma_{\Omega ESA}$
107	odb_esa_atmos_height	Height of the infrared atmosphere above Earth's surface when viewed by the each Earth sensor (km)	Real	2	16	TRW PCAD	4.3.2.3.5 (Simulate Earth Sensor Assembly)	H_{IR}
107A	odb_esa_phase_coeff_0	Zeroth order coefficient for ESA phase correction:; index = ESA number. (degrees)	Real	2	16	TRW	4.3.1.1.6	K_{p0}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
107B	odb_esa_phase_coeff_1	First order coefficient for ESA phase correction; index = ESA number. (degrees/degree)	Real	2	16	TRW	4.3.1.1.6	K_{p1}
107C	odb_esa_chord_coeff_0	Zeroth order coefficient for ESA chord correction; index = ESA number. (degrees)	Real	2	16	TRW	4.3.1.1.6	K_{c0}
107D	odb_esa_chord_coeff_1	First order coefficient for ESA chord correction; index = ESA number. (degrees/km)	Real	2	16	TRW	4.3.1.1.6	K_{c1}
107E	odb_esa_chord_coeff_2	Second order coefficient for ESA chord correction; index = ESA number. (degrees/km^2)	Real	2	16	TRW	4.3.1.1.6	K_{c2}
107F	odb_esa_blank_region	Angles specifying the ESA blanking region, defined such that the center of the unblanked region is 0.0; index: 1 = blanking region angle, (1 = region start angle, 2 = region end angle); 2 = ESA (degrees, range is -180 to 180.0)	Real	4	32	TRW PCAD	4.1.3.5 (Position Target on Optical Axis)	θ_w
	ODE_FSS_Characteristics		**Variable**		**80**			
108	Deleted							
109	Deleted							
110	Deleted							

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
111	Deleted							
112	Deleted							
113	Deleted							
114	odb_fss_beta_coeffs	Calibration coefficients for the angular measurement about the FSS y-axis, for each FSS; index: 1 = FSS, 2 = calibration coefficients	Real	3x2	48	TRW PCAD	4.3.1.1.5, 4.3.1.4, 4.3.2.3.4 (Process FSS data, Simulate FSS)	A_1, A_2, A_3
115	odb_fss_alpha_coeffs	Calibration coefficients for the angular measurement about the FSS x-axis, for each FSS; index: 1 = FSS, 2 = calibration coefficients	Real	3x2	48	TRW PCAD	4.3.1.1.5, 4.3.1.4, 4.3.2.3.4 (Process FSS data, Simulate FSS)	B_1, B_2, B_3
116	odb_fss_nom_align	Defines the rotational transformation from the nominal FSS frame to the AXAF body frame, for each FSS; index: 3 = FSS	Real	3x3x2	144	TRW PCAD	4.3.1.1.5, 4.3.1.3, 4.3.1.4, 4.3.2.3.4 (Process FSS data, Determine FSS Alignment, Simulate FSS)	M_{mF} M_{mS}
117	odb_fss_misalign	Defines the rotational transformation from the misaligned FSS frame to the nominal FSS frame, for each FSS; index: 3 = FSS	Real	3x3x2	144	TRW PCAD	4.3.1.1.5, 4.3.1.3, 4.3.1.4, 4.3.2.3.4 (Process FSS data, Determine FSS Alignment, Simulate FSS)	$M_{FF'}$ $M_{SS'}^{old}$
118	Deleted							
119	Deleted							
120	Deleted							

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
121	Deleted							
122	Deleted							
123	Deleted							
124	Deleted							
125	Deleted							
126	Deleted							
127	Deleted							
128	Deleted							
	ODE_IRU_Characteristics		**Variable**		**80**			
129	odb_iru_bias_error	IRU bias estimate error. (arcseconds/second)	Real		8	TRW PCAD	4.3.1.5 (Determine IRU Calibration Parameters)	N/A
130	odb_iru_channel_map	Map of IRU channel select to the IRU channels represented. IRU channel select is a telemetry value.	Integer	4x126	2016	TRW PCAD	4.3.1.13 (Convert IRU Counts to Spacecraft Body Rates)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
131	odb_iru_scale_fact_align	IRU scale factor alignment matrix for each channel combination and high and low rate mode; index: 1 = row, 2 = column, 3 = channel select, 4 = high/low rate mode. (=1, high rate mode data; =2, low rate mode data). Each 3x4 matrix specifies the alignment from the specified set of IRU channels for index 3 (see odb_iru_channel_map) to the Aspect Camera Assembly frame. For matrices corresponding to IRU combinations which include only three channels, the elements of the fourth column must be zero. (unitless)	Real	3x4x126x2	24192	CSC	4.3.1.1.3, 4.3.1.5 (Convert IRU Counts to Spacecraft Body Rates, Determine IRU Calibration Parameters)	G
132	odb_iru_drift_rate	IRU drift rate bias vector for each channel combination and high and low rate mode expressed in the Aspect Camera Assembly frame; index: 1 = vector element, 2 = channel select, 3 = high/low rate mode. (=1, high rate mode data; =2, low rate mode data). (degrees/second)	Real	3x126x2	6048	CSC	4.3.1.1.3, 4.3.1.5 (Convert IRU Counts to Spacecraft Body Rates, Determine IRU Calibration Parameters)	b

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
133	odb_iru_low_rate_scales	Low rate mode scale factor coefficients for the eight IRU output channels as defined in the description of odb_iru_channel_map; index: 1 = coefficient, 2 = positive/negative delta counts, 3 = channel (=1, negative counts scales; =2, positivie counts scales). These values must correspond to the sample period with which the data is telemetered.	Real	4x2x8	512	TRW PCAD	4.3.1.13 (Convert IRU Counts to Spacecraft Body Rates)	C_n^-, C_n^+
134	odb_iru_high_rate_scales	High rate mode scale factors for the eight IRU output channels as defined in the description of odb_iru_channel_map. These values must correspond to the sample period with which the data is telemetered. (radians/second/count)	Real	8	64	TRW PCAD	4.3.1.13 (Convert IRU Counts to Spacecraft Body Rates)	C_{hi}
135	odb_iru_sample_period	IRU telemetered counts sample period (seconds)	Relative Time		8	TRW PCAD	4.3.1.5, 4.3.2.3.1 (Determine IRU Calibration Parameters, Simulate IRU)	τ_g
136	odb_iru_large_angle_error	Angular error associated with IRU calibration uncertainty applied across a 90 degree maneuver (arcseconds)	Real		8	TRW PCAD	4.3.1.5 (Determine IRU Calibration Parameters)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
137	odb_iru_max_count	Total number of counts (states) of the IRU pulse accumulator (counts)	Integer		4	TRW PCAD	4.3.1.1.2 (Gap Fill IRU data)	σ_r
	ODE_LGA_Characteristics		**Variable**		**80**			
138	odb_lga1_or	Joining point of LGA-1 to spacecraft body in the AXAF body frame (meters).	Real	3	24	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage), 4.1.8.3 (Calculate Sunshade Door Blockage, Calculate SA Blockage)	$\check{P}$
139	odb_lga2_or	Joining point of LGA-2 boresight to spacecraft body in the AXAF body frame (meters).	Real	3	24	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage), 4.1.8.3 (Calculate Sunshade Door Blockage, Calculate SA Blockage)	$\check{P}$
140	odb_lga1_maxang	Maximum allowable angle between LOS vector to DSN and the LGA-1 boresight vector (degrees).	Real		8	TRW PCAD	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	N/A
141	odb_lga2_maxang	Maximum allowable angle between LOS vector to DSN and the LGA-2 boresight vector (degrees).	Real		8	TRW PCAD	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	N/A
142	odb_lga1_yaw	Yaw offset of LGA-1 in the AXAF body frame (degrees).	Real		8	TRW MDI	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	y_i

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
143	odb_lga1_pitch	Pitch offset of LGA-1 in the AXAF body frame (degrees).	Real		8	TRW MDI	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	p_i
144	odb_lga2_yaw	Yaw offset of LGA-2 in the AXAF body frame (degrees).	Real		8	TRW MDI	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	y_i
145	odb_lga2_pitch	Pitch offset of LGA-2 in the AXAF body frame (degrees).	Real	3	24	TRW MDI	4.1.8.1 (Calculate LGA-DSN Station Line-of -Sight)	P
146	odb_lga1_trans	Identifier for transponder used with LGA 1; =1, transponder 1, =2, transponder 2	Integer		4	TRW SE	N/A	
147	odb_lga2_trans	Identifier for transponder used with LGA 2; =1, transponder 1, =2, transponder 2	Integer		4	TRW SE	N/A	
	ODE_SI_Characteristics		**Variable**		**80**			
148	odb_acisi_fids	Defines the nominal and alternate fid light sets for ACIS-I in order of preference (index: 1 = fid light, 2 = fid light set)	Integer	3x5	60	ASC	4.1.3.12.n4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
149	odb_aciss_fids	Defines the nominal and alternate fid light sets for ACIS-S in order of preference (index: 1 = fid light, 2 = fid light set)	Integer	3x8	96	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A
150	odb_hrci_fids	Defines the nominal and alternate fid light sets for HRC-I in order of preference (index: 1 = fid light, 2 = fid light set)	Integer	3x4	48	ASC	4.1.3.12.n(Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A
151	odb_hrcs_fids	Defines the nominal and alternate fid light sets for HRC-S in order of preference (index: 1 = fid light, 2 = fid light set)	Integer	3x4	48	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A
152	odb_hrc_overshoot	Defines the amount past the requested HRC SIM position the SIM should be commanded before returning to the requested HRC SIM position in order to compensate for hysteresis (SIM motor steps; index: 1 = overshoot for SIM focus positions greater than the snapover point, 2 = overshoot for SIM focus positions less than the snapover point)	Real	2	16	TRW SE	4.1.5.1 (Calculate SI Configuration Parameters)	$X_{h,i}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
153	odb_acis_overshoot	Defines the amount past the requested ACIS SIM position the SIM should be commanded before returning to the requested ACIS SIM position in order to compensate for hysteresis (SIM focus motor steps; index: 1= overshoot for SIM focus positions greater than the snapover point, 2 = overshoot for SIM focus positions less than the snapover point)	Real	2	16	TRW SE	4.1.5.1 (Calculate SI Configuration Parameters)	$X_{h,i}$
153A	odb_snap_over	Defines the snapover point for SIM focus positioning (SIM focus motor steps)	Real		8	TRW SE	4.1.5.1 (Calculate SI Configuration Parameter)	X_S
153B	odb_ranget	Defines the allowed range for SIM translation motion (SIM translation motor steps; index: 1 = minimum allowed translation position, 2 = maximum allowed translation position)	Real	2	8	TRW SE	4.1.5.1 (Calculate SI Configuration Parameter)	$X_{t,\min}$, $X_{t,\max}$
153C	odb_rangef	Defines the allowed range for SIM focus motion (SIM focus motor steps; index: 1 = minimum allowed focus position, 2 = maximum allowed focus position)	Real	2	8	TRW SE	4.1.5.1 (Calculate SI Configuration Parameter)	$X_{f,\min}$, $X_{f,\max}$
154	odb_acisbias_dur	Defines the duration of an ACIS bias measurement (seconds)	Real		8	ASC	4.1.5 (Support SI Configuration)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
155	odb_si_align	Defines the rotational transformation from the science instrument (SI) frame to the AXAF body frame for each SI; index: 3=SI, (1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S)	Real	3x3x4	288	TRW MASS PROP	4.1.3.2, 4.1.3.5 (Calculate Nominal Roll, Position Target on Optical Axis)	$[M_j]$
156	odb_si_simt	Science instrument module (SIM) position for nominal SI translation position for each SI (SIM translation motor steps); index: 1=SI, (1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S)	Integer	4	16	TRW SE/SI	4.1.5 (Support SI Configuration)	Z_n
157	odb_si_simv	Science instrument module (SIM) position for nominal SI focus position for each SI (SIM focus motor steps); index: 1=SI, (1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S)	Integer	4	16	TRW SE/SI	4.1.5 (Support SI Configuration)	X_n
158	odb_si_fovy	Minimum and maximum SI FOV extent angles (size) in the y measurement direction (rotation about the z-axis); index: 1=angle, 1=minimum, 2=maximum; 2=SI: 1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S) (degrees)	Real	2x4	64	TRW SE/SI	4.1.5 (Support SI Configuration)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
159	odb_si_fovz	Minimum and maximum SI FOV extent angles (size) in the z measurement direction (rotation about the y-axis); index: 1=angle, 1=minimum, 2=maximum; 2=SI: 1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S) (degrees)	Real	2x4	64	TRW SE/SI	4.1.5 (Support SI Configuration)	N/A
160	odb_si_ratet	Translation move rate for repositioning SIM (SIM translation motor steps per second)	Real		8	TRW SI	4.1.5 (Support SI Configuration)	r_{TSC}
161	odb_si_ratef	Focus move rate for vertical repositioning of the SIM (SIM focus motor steps per second)	Real		8	TRW SI	4.1.5 (Support SI Configuration)	r_{FA}
162	odb_si_settlet	Translation motion settling time (seconds)	Real		8	TRW SI	4.1.5 (Support SI Configuration)	N/A
163	odb_se_settlef	Vertical focus motion settling time (seconds)	Real		8	TRW SI	4.1.5 (Support SI Configuration)	N/A
164	odb_acisi_fidpos	Defines the locations of the 6 ACIS FID lights with respect to the ACIS-I detector coordinate system origin; (meters; index: 1 = y and z coordinate, 2 = FID light)	Real	2x6	64	TRW MDI	4.1.3.12.1, 4.3.1.5, 4.3.2.3.6.2	Y_s, Z_s
164A	odb_aciss_fidpos	Defines the locations of the 6 ACIS FID lights with respect to the ACIS-S detector coordinate system origin; (meters; index: 1 = y and z coordinate, 2 = FID light)	Real	2x6	64	TRW MDI	4.1.3.12.1, 4.3.1.5, 4.3.2.3.6.2	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
165	odb_hrci_fidpos	Defines the locations of the 4 HRC-I FID lights with respect tot he HRC-I detector coordinate system origin; index: 1 = y and z coordinate, 2 = FID light (meters)	Real	2x4	64	TRW MDI	4.1.3, 4.3.1.7, 4.3.2.3.6.2	Y_s, Z_s
166	odb_hrcs_fidpos	Defines the locations of the 4 HRC-S FID lights with respect to the HRC-S detector coordinate system origin; index: 1 = y and z coordinate, 2 = FID light (meters)	Real	2x4	64	TRW MDI	4.1.3, 4.3.1.7, 4.3.2.3.6.2	Ys, Zs
167	odb_tsc_steps	Defines the conversion factor from TSC steps to meters (meters/TSC step)	Real		8	TRW SI	4.1.3, 4.3.1.7, 4.3.2.3.6.2	Zc
168	odb_fa_numsteps	Defines the number of entries in the FA step conversion look up table, odb_fa_steps	Integer		4	TRW SI		N/A
169	odb_fa_steps	Defines a lookup table for converting the FA position in steps to the FA position in meters; index:1=position in FA motor steps, 2 = position in meters	Real	2x3000	48000	TRW SI	4.1.3, 4.3.1.7, 4.3.2.3.6.2	Xci

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
169A	odb_hrc_dither_pad	Defines the time pad prio to the end of the spacecraft maneuver at which HRC voltage must be turned off (relative time)	ASCII		17	ASC	4.1.11 (Expand OR and ER)	N/A
169B	odb_obscal_idlim	Defines the division between the OBS and CAL IDs generated by the ASC and the OFLS MPS. The ASC may use numbers between 0 and this number. The MPS may use numbers between this number and 65536.	Integer		4	ASC	N/A	
169C	odb_siwarm_dur	Defines the time required before an SI can be used after the instrument has been off (e.g., after power down during eclipse) (seconds; index = SI)	Real	2	16	ASC	N/A	
169D	odb_grating_dur	Defines the time required to move a single grating into or out of the focal plane (seconds)	Real		8	TRW	N/A	
169E	odb_hrccnfg_pad	Defines the time required to reconfigure the HRC between different operational configurations (seconds)	Real		8	ASC	N/A	
172	odb_sa_edge_sep	Distance from the AXAF body to the closest SA edge for each SA (meters)	Real	2	16	TRW MDI	4.1.8.3 (Calculate SA blockage)	S_1 , S_2

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
173	odb_sarawout	Defines raw output for each SA (watts)	Real	2	16	TRW MDI	4.1.6.2 (Estimate SA Power Output)	F_{1_1}, F_{1_2}
174	odb_sa_min_current	Minimum solar array current for each SA when the Sun is on the active (-z) side of the solar arrays (amperes)	Real	2	16	TRW PCAD	4.3.1.1.4 (Process Coarse Sun Sensor Data)	I_{thresh} in item c.
175	odb_sa_i_noise_mean	Mean of measurement noise on the SA current measurement for each SA (amperes)	Real	2	16	TRW PCAD	4.3.2.3.2 (Simulate SA)	μ_{ISA}
176	odb_sa_i_noise_std_dev	Standard deviation of measurement noise on the SA current measurement for each SA (amperes)	Real	2	16	TRW PCAD	4.3.2.3.2 (Simulate SA)	σ_{ISA}
177	odb_sa_angle_noise_mean	Mean of measurement noise on the SA resolver measurements for each SA (degrees)	Real	2	16	TRW PCAD	4.3.2.3.2 (Simulate SA)	$\mu\sigma_{SA1}$, $\mu\sigma_{SA2}$
177	odb_sa_angle_noise_mean	Mean of measurement noise on the SA resolver measurements for each SA (degrees)	Real	2	16	TRW PCAD	4.3.2.3.2 (Simulate SA)	$\mu\theta_{SA1}$, $\mu\theta_{SA2}$
178	odb_sa_angle_noise_std_dev	Standard deviation of measurement noise on the SA resolver measurements for each SA (degrees)	Real	2	16	TRW PCAD	4.3.2.3.2 (Simulate SA)	$\sigma\theta_{SA1}$, $\sigma\theta_{SA2}$
179	odb_sa_center_mass	Defines the vector from the spacecraft center of mass to the SA center of radiation pressure in the AXAF body frame for each SA	Real	3x2	48	TRW MDI	4.1.3.12.1 (Calculate S/C Momentum during maneuvers) (calculate S/C Momentum during maneuvers)	$\vec{R}_{sa_h}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
180	odb_sa_diff_ref	Defines the solar radiation pressure coefficient of diffuse reflection for each SA	Real	2	16	TRW PCAD	4.1.3.12.1 (Calculate S/C Momentum during maneuvers) (calculate S/C Momentum during maneuvers)	C_{d_h}
181	odb_sa_spec_ref	Defines the solar radiation pressure coefficient of specular reflection for each SA	Real	2	16	TRW PCAD	4.1.3.12.1 (Calculate S/C Momentum during maneuvers) (calculate S/C Momentum during maneuvers)	C_{s_h}
181A	odb_sa_offpoint	Defines the value by which the solar arrays are to be off-pointed during shadow processing to artificially extend the shadow duration (degrees: index = SA)	Real	2	16	TRW	To Be Provided	
181B	odb_sa_offpoint_when	Defines when SA off-pointing can be performed: allowed values are “BEFORE”, “AFTER”, “EITHER”	ASCII		6	TRW	To Be Provided	
	ODE_SSR_Characteristics		**Variable**		**80**			
182	odb_record_rate	SSR record rate (bytes per second)	Integer		4	TRW CCDM	4.1.9.1 (Calculate Current Recorder Usage)	r
183	odb_plybk_rate	SSR playback rate (bytes per second; index: SSR)	Integer	2	4	TRW CCDM	4.1.9.2 (Check Playback Support Requirements)	N/A
184	odb_blocks	Number of blocks in the SSR (index: SSR)	Integer	2	4	TRW CCDM	4.1.9.2, 4.1.11 (Check playback support requirements) (Check DOT Constraints)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
185	odb_bytes	Number of bytes per block in the SSR	Integer		4	TRW CCDM	4.1.11 (Check DOT Constraints)	N/A
186	odb_ssr_id	Identification of operational SSR (1=SSR-1, 2=SSR-2)	Integer		4	TRW CCDM	4.1.9.2 (Check Playback Support Requirements)	N/A
186	odb_ssr_id	Identification of operational SSR (1=SSR-1, 2=SSR-2)	Integer		4	TRW CCDM	4.1.9.2 (Check Playback Support Requirements)	To Be Provided
	ODE_Spacecraft_Characteristics		**Variable**		**80**			
187	odb_boresight	HRMA boresight vector in the AXAF body frame	Real	3	24	TRW MDI	4.1.8.4 (Calculate S/C body blockage)	N/A
188	odb_cylrad	Cylindrical radius of the spacecraft measured from the AXAF body frame x-axis (meters)	Real		8	TRW MDI	4.1.8.4 (Calculate S/C body blockage)	W
189	odb_body_length	Length of spacecraft (meters)	Real		8	TRW MDI	4.1.8.4 (Calculate S/C body blockage)	L
190	odb_mom_inr_sc	Defines moment of inertia tensor of the spacecraft in the AXAF body frame; index:3=SIM positions (1=ACIS, 2=HRC, 3=CENTER)	Real	3x3x3	216	TRW MASS PROP	4.1.3.12.1 (Calculate S/C Momentum during maneuvers)	I_{sc}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
191	odb_mom_inr_p	Defines moment of inertia tensor for the spacecraft in the frame defined by the principle momentum axes; index: 3=SIM position (1=ACIS, 2=HRC, 3=CENTER)	Real	3x3x3	216	TRW MASS PROP	4.1.3.12.2 (Calculate S/C Momentum at Fixed Attitude)	I_{pm}
192	odb_sc_to_pma	Defines the rotation matrix from the AXAF body frame to the frame defined by the principle momentum axes; index: 3 = SIM position (1=ACIS, 2=HRC, 3=CENTER)	Real	3x3x3	216	TRW MASS PROP	4.1.3.12.2 (Calculate S/C Momentum at Fixed Attitude)	M_m
193	odb_plane_array	Defines plane array for decomposition of spacecraft geometry; index: 1 =plane data (1 to 3 = vector from center of spacecraft mass to center of pressure for the ith plane, 4 - coefficient of diffuse reflection, 5=coefficient of specular reflection, 6= surface area (meters2), 7 to 9=surface normal unit vector); 2 = plane	Real	9x10	720	TBD	4.1.3.12.1 (Calculate S/C Momentum during maneuvers)	R_{pi}, C_{si}, C_{di}, N_i

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
194	odb_cylinder_array	Defines cylinder array for decomposition of spacecraft geometry; index: 1 = cylinder data (1 to 3 = vector from center of spacecraft mass to center of pressure for the ith cylinder, 4 - coefficient of diffuse reflection, 5=coefficient of specular reflection, 6= cylinder radius (meter), 7 = cylinder height, 8 to 10 = vector for symmetry axis in spacecraft coordinates); 2 = cylinder	Real	10x10	800	TBD	4.1.3.12.1 (Calculate S/C Momentum during maneuvers)	R_{ck}, C_{sk},C_{dk}, r_k, h_k
195	odb_point_error	Defines the maximum expected pointing error during attitude hold (arcseconds)	Real		8	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
196	odb_n_slew_ang	Defines the number of angles in the slew angle table, odb_slew_ang	Integer		4	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
197	odb_ slew_ang	Defines the slew angle range limits for the lookup table for maximum pointing error at the end of a maneuver, odb_slew_err. Values must be in increasing order, an initial value of 0 is assumed. (degrees)	Real	10	80	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
198	odb_slew_err	Defines the maximum pointing error at the end of a maneuver for ranges in slew angle defined in odb_slew_ang (arcseconds)	Real	10	80	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
198A	odb_num_pitch_angles	Defines the number of solar pitch angles defined in the array of solar pitch angle bins, odb_solar_pitch. Value must be greater than or equal to 3.	Integer		4	TRW PCAD	4.1.3.13.1.1.2 Calculate the Solar Radiation Torque Based on the ODB-Specified Solar Flux Torque Tables	N_β
198B	odb_solar_pitch_angles	Defines each solar pitch angle bin used in the solar torque lookup table, odb_solar_torque. Bin 1 is assumed to extend from 0 to the value of array element 1. The last bin is assumed to go from the value of the odb_num_pitch_angles-th bin to 180 (degrees; index: pitch angle bin)	Real	60	480	TRW PCAD	4.1.3.13.1.1.2 Calculate the Solar Radiation Torque Based on the ODB-Specified Solar Flux Torque Tables	β
198C	odb_num_offnom_angles	Defines the number of off-nominal roll angles defined in the array of off-nominal roll angle bins, odb_offnom_roll_ angles. Value must be greater than or equal to 3. (index: solar pitch angle)	Integer	60	240	TRW PCAD	4.1.3.13.1.1.2 Calculate the Solar Radiation Torque Based on the ODB-Specified Solar Flux Torque Tables	N_ϕ

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
198D	odb_offnom_roll_angles	Defines each off-nominal roll angle bin used in the solar torque lookup table, odb_solar_torque. Array element (1,n) must be the maximum allowed negative off-nominal roll angle at any solar pitch angle defined in odb_solar_pitch_angle. The last array element (m,n) must be the maximum positive off-nominal roll at any solar pitch angle defined in odb_solar_pitch_angle (degrees; index: 1 = off-nominal roll, 2 = solar pitch angle)	Real	30x60	1800	TRW PCAD	4.1.3.13.1.1.2 Calculate the Solar Radiation Torque Based on the ODB-Specified Solar Flux Torque Tables	ϕ
198E	odb_solar_torque	Defines a table of values for the solar torque in terms of solar pitch angle and off-nominal roll angle (kilogram-meter^2/second^2; indices: 1=off-nominal roll bin, 2=solar pitch angle bin,)	Real	30x60	1800	TRW PCAD	4.1.3.13.1.1.2 Calculate the Solar Radiation Torque Based on the ODB-Specified Solar Flux Torque Tables	N_{SP}
199	odb_acis_to_hrma	Defines the distance from the ACIS fid light point of origin to the HRMA nodal point (meters)	Real	6	48	TRW/SE (ADS Group)	4.1.3.12.1(Calculate Fiducial Light Position)	R_H
200	odb_hrci_to_hrma	Defines the distance from the HRC-I fid light point of origin to the HRMA nodal point (meters)	Real	4	32	TRW/SE (ADS Group)	4.1.3.12.1(Calculate Fiducial Light Position)	R_H
201	odb_hrcs_to_hrma	Defines the distance from the HRCS fid light point of origin to the HRMA nodal point (meters)	Real	4	32	TRW (ADS Group)	4.1.3.12.1(Calculate Fiducial Light Position)	R_H

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
202	odb_nom_dither_amp	Defines the nominal dither amplitude for an observation (degrees) defined in terms of displacement of the ACA FOV in the direction of the y and z spacecraft body axes, listed in characteristics file in the following order: ACIS-I (z), ACIS-S (z), HRC-I (z), HRC-S (z), ACIS-I (y), ACIS-S (y), HRC-I (y), HRC-S (y) index: 1 = Science Instrument , 2 = direction (z,y) , where z denotes motion in pitch and y in yaw	Real	4x2	64	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
203	odb_nom_dither_freq	Defines the nominal dither frequency for an observation (degrees/second): defined in terms of displacement of the ACA FOV in the direction of the y and z spacecraft body axes, listed in the characteristics file in the following order: ACIS-I (z), ACIS-S (z), HRC-I (z), HRC-S (z), ACIS-I (y), ACIS-S (y), HRC-I (y), HRC-S (y) index: 1 = Science Instrument , 2 = direction (z,y) , where z denotes motion in pitch and y in yaw	Real	4x2	64	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
204	odb_nom_dither_ph	Defines the nominal dither phase for an observation (degrees) i defined in terms of displacement of the ACA FOV in the direction of the y and z spacecraft body axes, listed in characteristics file in the following order: ACIS-I (z), ACIS-S (z), HRC-I (z), HRC-S (z), ACIS-I (y), ACIS-S (y), HRC-I (y), HRC-S (y) index: 1 = Science Instrument , 2 = direction (z,y) , where z denotes motion in pitch and y in yaw	Real	4x2	64	ASC	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
205	odb_mom_unlcap	Defines the momentum unload capability of the MUPS and RCS along each of the 3 AXAF body axes (kilogram-meter^2/second) index: 1=x,y,z; 2 = MUPS, RCS	Real	3x2	48	TRW PCAD	4.1.3.13.3 (Predict S/C Momentum Dump Duration)	$H_{cap\ j,m}$
206	odb_mom_fp	Define firing period between pulses for thruster selection MUPS or RCS (seconds; index:1 = MUPS, RCS)	Real	2	16	TRW PCAD	4.1.3.13.3 (Predict S/C Momentum Dump Duration)	P_m
207	odb_unload_method	Defines the onboard method for momentum unloading; 1=MUPS, 2=RCS	Integer		4	TRW PCAD	4.1.3.13.3 (Predict S/C Momentum Dump Duration)	m
208	odb_mom_sign	Signed constants used to account for the differences in orientation between RCS and MUPS thrusters; index; 1= signed constants, 2 = thruster type	Integer	3x2	24	TRW PCAD	4.1.3.13.3 (Predict S/C Momentum Dump Duration)	$S_{gn\ j,m}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
208A	odb_mom_deadzone	Defines the dead zone around the commanded dump to momentum value used by the OBC when performing ground commanded momentum dumps (kilogram-meters^2/second)	Real	2	16	TRW PCAD	4.1.3.12.2 (Calculate S/C Momentum at Fixed Attitude)	M_m
209	odb_num_planes	The number of planes in the spacecraft model described by odb_plan_array	Integer		4	TBD	4.1.3.13.1 (Calculate S/C Momentum during maneuvers)	n_p
210	odb_num_cylinders	The number of cylinders in the spacecraft model described by odb_cylinder_array	Integer		4	TBD	4.1.3.13.1 (Calculate S/C Momentum during maneuvers)	n_k
211	odb_rw_dir	Defines the direction vector of each reaction wheel spin axis in the spacecraft body frame; index: 1=direction vector (x,y,z), 2 = reaction wheel	Real	3x6	144	TRW MDI	4.1.3.13.4 (Calculate Reaction Wheel Speeds)	A_{rw}
212	odb_rw_inr	Defines moment of inertia corresponding to each reaction wheel spin axis (kilogram-meters^2); index: reaction wheel	Real	6	48	TRW PCAD	4.1.3.13.4 (Calculate Reaction Wheel Speeds)	I_{rw}
213	odb_rw_bias_max	Defines the maximum reaction wheel bias speed for each reaction wheel (rad/sec); index: reaction wheel	Real	6	48	TRW PCAD	4.1.3.13.4 (Calculate Reaction Wheel Speeds)	$W_{B,MAX}$
214	odb_rw_bias_min	Defines the minimum reaction wheel bias speed for each reaction wheel (rad/sec); index: reaction wheel	Real	6	48	TRW PCAD	4.1.3.13.4 (Calculate Reaction Wheel Speeds)	$W_{B,MIN}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
214A	odb_prop_pulse	Defines the amount of propellant consumed for each MUPS thruster pulse (liters/pulse)	Real		8	TRW PCAD	4.1.3.13 (Manage Spacecraft Momentum)	N/A
214B	odb_spacecraft_mass	Mass of the spacecraft (kilograms)	Real		8	TRW	N/A	
214C	odb_spacecraft_area	Cross-sectional area of the spacecraft (kilometers^2)	Real		8	TRW	N/A	
	ODE_Sunshade_Characteristics		**Variable**		**80**			
215	odb_ssd_rot	Rotation angle of open sunshade door above the x-y plane measured from the +x-axis to the plane of the sunshade door in the +z direction (degrees)	Real		8	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage)	θ
216	odb_ssd_long	Maximum extent from the join point of the sunshade door to the edge of the sunshade door measured in the positive AXAF body frame x-direction if the sunshade door were in the AXAF body x-y plane (meters).	Real		8	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage)	L
217	odb_ssd_wide	Maximum extent in the AXAF body frame y-direction of the sunshade door (meters).	Real		8	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage)	W
218	odb_ssd_join	Joining point of sunshade door to spacecraft body in the AXAF body frame (meters).	Real	3	24	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage)	$\check{D}$

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Clock_Characteristics		**Variable**		**80**			
219	odb_obc_cnts_per_ro	The maximum VCDU count before rollover occurs	Integer		4	TRW PCAD	4.5.4.3, 4.5.4.4 (Convert GMT seconds to S/C Clock Cycles and Counts)	$N_{cnts/cyc}$
220	odb_obc_cnts_per_mc	The number of VCDU counts per OBC minor cycle	Real		8	TRW CCDM	4.5.4.3 (Convert GMT seconds to S/C Clock Cycles and Counts)	$N_{cnts/mc}$
221	odb_eph_base_ref	Defines the base reference time (OBC clock start time) for ephemeris time of perigee processing (HOSC GMT)	ASCII		21	TRW CCDM	4.5.3 (Estimate PCAD Ephemeris Coefficients)	t_{obc}
	ODE_Memory_Characteristics		**Variable**		**80**			
222	odb_obc_scp_mem	Defines areas of contiguous physical OBC stored command processor memory available for OFLS stored command loads; index: 1=starting address and the ending address; 2=memory area; 3 = OBC	Integer	2x256x2	256	TRW FSW	4.1.7 (Estimate OBC Memory Usage)	N/A
223	odb_obc_sw_mem	Defines areas of contiguous physical OBC software memory into which flight software updates can be loaded; index: 1=starting address and the ending address; 2=memory area; 3 = OBC	Integer	2x80x2	320	TRW FSW	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
224	odb_obc_tab_mem	Defines areas of contiguous physical OBC table memory into which OBC table loads can be loaded by the OFLS; index: 1=starting address and the ending address; 2=memory area; 3 = OBC	Integer	2x200x2	800	TRW FSW	N/A[2]	
225	odb_cpe_sw_mem	Defines areas of contiguous CPE software memory into which CPE flight software updates can be loaded; index: 1=starting address and the ending address; 2=memory area; 3 = CPE	Integer	2x16x2	256	TRW FSW	N/A[2]	
226	odb_cpe_tab_mem	Defines areas of contiguous CPE table memory into which CPE tables can be loaded by the OFLS; index: 1=starting address and the ending address; 2=memory area; 3 = CPE	Integer	2x16x2	256	TRW FSW	N/A[2]	
227	odb_ac_sw_mem	Defines areas of contiguous AC software memory into which AC flight software updates can be loaded; index: 1=starting address and the ending address; 2=memory area; 3 = AC	Integer	2x16x2	256	TRW/ BALL	N/A[2]	
228	odb_ac_tab_mem	Defines areas of contiguous AC table memory into which AC tables can be loaded by the OFLS; index: 1=starting address and the ending address; 2=memory area; 3 = AC	Integer	2x16x2	256	TRW/ BALL	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
229	odb_iuee_sw_mem	Defines areas of contiguous IU EEPROM software memory into which IU EEPROM flight software updates can be loaded; index: 1=starting address and the ending address; 2=memory area; 3 = IU EEPROM	Integer	2x16x2	256	TRW FSW	N/A[2]	
230	odb_sim_sw_mem	Defines areas of contiguous SIM software memory into which SIM flight software updates can be loaded; index: 1=starting address and the ending address; 2=memory area; 3=SIM	Integer	2x16x2	256	TBD		
231	odb_ephin_sw_mem	Defines areas of contiguous ephin software memory into which ephin flight software updates can be loaded; index: 1= starting address and the ending address; 2=memory area; 3= ephin	Integer	2x16x2	256	TBD		
232	odb_ctuee_sw_mem	Defines areas of contiguous CTU EEPROM software memory into which CTU EEPROM flight software updates can be loaded; index: 1 = starting address and the ending address; 2 = memory area; 3 = CTU EEPROM	Integer	2x16x2	256	TRW FSW		
233	odb_obc_num_ats_scs	Defines the number of SCSs available for mapping ATS commands	Integer		4	TRW FSW	4.2.2.2 (Map ATS Commands to Memory)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
234	odb_obc_first_ats_scs	Defines the SCS number of the first SCS available for mapping ATS commands	Integer		4	TRW FSW	4.2.2.2 (Map ATS Commands to Memory)	N/A
235	odb_obc_size_ats_scs	Defines size of each SCS available for mapping ATS commands, up to the number of ATS SCSs identified by odb_obc_num_ats_scs (SCS steps)	Integer	256	1024	TRW FSW	4.2.2.2 (Map ATS Commands to Memory)	N/A
236	odb_obc_num_rts_scs	Defines the number of SCSs available for mapping RTS commands	Integer		4	TRW FSW	4.2.2.2 (Map RTS Commands to Memory)	N/A
237	odb_obc_first_rts_scs	Defines the SCS number of the first SCS available for mapping RTS commands	Integer		4	TRW FSW	4.2.2.2 (Map RTS Commands to Memory)	N/A
238	odb_obc_size_rts_scss	Defines size of each SCS available for mapping RTS commands, up to the number of RTS SCSs identified by odb_obc_num_rts_scs (SCS steps)	Integer	256	1024	TBD	4.2.2.2 (Map RTS Commands to Memory)	N/A
238A	odb_max_scsblk_cmd	Defines the maximum number of commands (steps) in an SCS load block	Integer		4	TRW	N/A	
238B	odb_max_memblk_word	Defines the maximum number of words in an absolute memory load block	Integer		4	TRW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Resource_Characteristics		**Variable**		**80**			
239	odb_num_resource	Number of resources scheduled by OFLS. Maximum number is 20.	Integer		4	CSC	4.1.1.3 (Build Mission Schedule Resource Definition)	N/A
240	odb_res_name1	Name of first resource in the pair	ASCII	20	200	CSC	4.1.1.3 (Resource Definition)	N/A
241	odb_res_name2	Name of second resource in the pair	ASCII	20	200	CSC	4.1.1.3 (Resource Definition)	N/A
242	odb_type	Type of relation between the pair of resources. I- indicates incompatible; Is indicates resources incompatible for scheduling, however operationally valid; C indicates compatible; Cnp indicates compatible resources, however not preferred; R indicates required combination; R1 indicates that only one of the resource combination is required.	ASCII	20x20	1200	CSC	4.1.1.3 (Resource Definition)	N/A
243	odb_sep_time	Minimum separation between requests using the resources (relative time).	ASCII	20x20	6,800	CSC	4.1.1.3 (Resource Definition)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
244	odb_sep_type	Separation type; 1=separation between start times; 2=separation between the end of the first in the pair and the start of the second in the pair).	Integer	20x20	1,600	CSC	4.1.1.3 (Resource Definition)	N/A
245	odb_cmd_words	Number of command words required to perform activity using specified resource	Integer	20	80	CSC	4.1.7 (Estimate OBC Memory Usage)	ΔC_i
	ODE_Telemetry_Characteristics		**Variable**		**80**			
246	odb_maj_frame_length	The major frame length in minor frames (index: telemetry format).	Integer	8	32	TRW CCDM	4.5.4.2 (Perform GMT Consistency Check)	F_{len}
247	odb_min_frame_period	The minor frame period (seconds; index: telemetry format)	Real	8	64	TRW CCDM	4.5.4.2 (Perform GMT Consistency Check)	f_{prd}
248	odb_sub_frame_length	The number of minor frames per HOSC packet (transfer frame).	Integer	8	32	TRW CCDM	4.5.4.2 (Perform GMT Consistency Check)	l_s
249	odb_min_frame_uncertainty	The uncertainty (maximum error magnitude) in the minor frame period (seconds; index: telemetry format).	Real	8	64	TRW CCDM	4.5.4.2 (Perform GMT Consistency Check)	U
250	odb_axaf_internal_delay	The internal AXAF time delay (seconds; index=telemetry rate (256, 512, 1024)).	Real	3	24	TRW CCDM	4.4.3.1.3 (Compute Total Adjustment to Ground receive Time)	$D_{axaf\text{-}fmt}$
251	odb_bit_rate	Bit rate (bits per second; index: telemetry format)	Integer	8	32	TRW CCDM	4.4.3.1.1 (Computer Adjustment to ground receive time for frame offset)	B_r

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
252	odb_telemetry_format	Defines the identifier by which the telemetry format is identified in the telemetry stream (index: telemetry format)	ASCII	8	24	TRW CCDM	4.4.3.1.2 (Computer Total Adjustment to Ground Receive Time)	N/A
253	odb_sync_offset	Defines the number of bits between the first bit of the attached synch marker and the first bit in the VCDU counter (seconds; index: telemetry format)	Integer	8	32	TRW CCDM	4.4.3.1.2 (Computer Total Adjustment to Ground Receive Time)	B_r
254	odb_aca_latency	Constant amount of time that ACA data is delayed between the time it is observed and the time it is put into telemetry (seconds)	Real		8	TRW CCDM	N/A	N/A
255	odb_css_latency	Constant time delay between the beginning of the minor frame in which the CSS data is reported and the time at which the data was sampled. Note that the delay may be negative. (seconds)	Real		8	TRW CCDM	N/A	N/A
256	odb_fss_latency	Constant time delay between the beginning of the minor frame in which the FSS data is reported and the time at which the data was sampled. Note that the delay may be negative. (seconds)	Real		8	TRW CCDM	N/A	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
257	odb_esa_latency	Constant time delay between the beginning of the minor frame in which the ESA data is reported and the time at which the data was sampled. Note that the delay may be negative. (seconds)	Real		8	TRW CCDM	N/A	N/A
258	odb_iru_latency	Constant time delay between the beginning of the minor frame in which the IRU data is reported and the time at which the data was sampled. Note that the delay may be negative. (seconds)	Real		8	TRW CCDM	N/A	N/A
259	odb_obc_latency	Constant time delay between the beginning of the minor frame in which the OBC data is reported and the time at which the data was sampled. Note that the delay may be negative. (seconds)	Real		8	TRW CCDM	N/A	N/A
	ODE_DSN_Characteristics		**Variable**		**80**			
260	odb_gs_long	Planetocentric longitudes of GSTDN/DSN stations (degrees).	Real	20	160	TRW SE	4.5.2.4, 4.5.4.18 (Determine DSN Coverage, Calculate the Position of a DSN Station in GCI Coordinates)	ϕ
261	odb_gs_distance	The distance from the earth center to the ground station in kilometers.	Real	20	160	TRW SE	4.1.8.1 (Calculate LGA-DSN Station Line-of-Sight)	R_D

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
262	odb_gs_lat	Planetocentric latitudes of GSTDN/DSN stations (degrees).	Real	20	160	TRW SE	4.5.2.4, 4.5.4.18 (Determine DSN Coverage, Calculate the Position of a DSN Station in GCI Coordinates)	θ_D
263	odb_gs_name	GSTDN/DSN mnemonics.	ASCII	20	120	TRW SE	4.5.2.4, 4.1.8.5 (Determine DSN Coverage, Select Optimal DSN Station and LGA)	N/A
263A	odb_gs_code	GSTDN/DSN antenna id code supplied in the DSN telemetry	Integer	20	80	TRW	4.4.3.1.2 (Compute Adjustment to Ground Receive Time for Propagation Delays)	N/A
264	odb_dsn_internal_delay	Internal time delay at DSN (seconds; index: DSN station)	Real	20	160	TRW SE	4.4.3.1.3 (Compute Total Adjustment to Ground Receive Time)	$D_{dsn\text{-}fmt}$
265	odb_dsn_return_services_delay	DSN return services time delay (seconds; index: DSN station)	Real	20	160	TRW SE	4.4.3.1.3 (Compute Total Adjustment to Ground Receive Time)	D_{dsn}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
266	odb_soe_codes	Sequence of events (SOE) code in the DSN schedule identifying the activities to be performed for the scheduled communication contact; Index: 1=SOE Code (1 alphanumeric character), 2=uplink (YES or NO), 3=downlink (YES or NO), 4=ranging (YES or NO), 5=bit rate (1024, 512, 256,32), 6=bit rate change (1024, 512, 256,32,none), 7=bit rate change time (MM minutes relative to BOT in the DSN schedule), 8=recorder dump start time (MM minutes relative to BOT in the DSN schedule), 9=recorder dump stop time (MM minutes relative to BOT in the DSN schedule),	ASCII	9x36	1296	TBD		
267	odb_wrk_codes	Defines the work category in the DSN schedule denoting the contact activity; index: 1=work code, 2=activity description	ASCII	2x5	70	TBD		
268	odb_dsn_spacecraft_name	Defines the spacecraft name used in the DSN schedule for AXAF	ASCII		4	TBD		

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
268A	odb_gs_mask	Defines the ground station visibility in terms of azimuth and elevation angles. The n^{th} elevation angle applies for azimuth angles greater than the n^{th} azimuth angle and less than or equal to the $(n+1)^{th}$ azimuth angle. (degrees; index: 1 = azimuth angle bin, 2 = elevation angle and azimuth angle, 3 = DSN station)	Real	36x2x20	11520	TRW	4.5.2.4 Determine DSN Coverage	λ_A
	ODE_Power_Characteristics		**Variable**		**80**			
269	odb_cc_slope	The slope battery constant from the calibration curves in ampere-hours per psi.	Real	3	24	MSFC	4.4.1.7 (Compute Battery Capacity)	b(1)
270	odb_cc_intercept	The y-intercept battery constant from the calibration curves in psi.	Real	3	24	MSFC	4.4.1.7 (Compute Battery Capacity)	b(0)
271	odb_calint	Defines maximum time interval between power calculations (minutes).	Real		8	TBD	4.1.6.1 (Estimate Power Load)	N/A
272	odb_therm	Defines thermal power delta lookup table based on Sun angle in the AXAF body x-z plane (watts; index: 10 degree bins for Sun angle from minimum allowed Sun angle to maximum allowed Sun angle)	Real	25	200	TBD	4.1.6.1 (Estimate Power Load)	Δ_{TH}

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
273	odb_maxangsun	Defines maximum Sun angle for thermal power delta lookup table, odb_therm (degrees; index 1=sunshade door open, 2=sunshade door closed)	Real	2	16	TBD	4.1.6.1 (Estimate Power Load)	N/A
274	odb_ambpl	Defines ambient power load (watts).	Real		8	TBD	4.1.6.1 (Estimate Power Load)	P_A
275	odb_epslos	Defines EPS loss factor.	Real		8	TBD	4.1.6.1 (Estimate Power Load)	L
276	odb_volfac	Defines voltage factor.	Real		8	TBD	4.1.6.2 (Estimate Solar Array Power Output)	F_3
277	odb_suninf	Defines sun intensity factor.	Real		8	TBD	4.1.6.2 (Estimate Solar Array Power Output)	F_2
278	odb_chargecb	Defines charge cutback (watts).	Real		8	TBD	4.1.6.2 (Estimate Solar Array Power Output)	F_4
279	odb_bateff	Defines battery efficiency factor.	Real		8	TBD	4.1.6.3 (Estimate Predicted Power State)	B_E
280	odb_batvolt	Defines battery voltage (volts).	Real		8	TBD	4.1.6.3 (Estimate Predicted Power State)	V
281	odb_batcap	Defines battery capacity (ampere-hours).	Real	3	24	TRW EPS	4.1.6.3, 4.4.1.5 (Estimate Predicted Power State, Compute Depth of Discharge)	B_C B_{nc}
282	odb_pwr_man	Power load during a spacecraft maneuver (watts)	Real		8	TBD	4.1.6.1 (Estimate Power Load)	P_X
283	odb_pwr_ac	Power load for the aspect camera (watts)	Real		8	TBD	4.1.6.1 (Estimate Power Load)	P_X

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
284	odb_pwr_sa	Power load during a solar array maneuver (watts; index = SA)	Real	2	16	TBD	4.1.6.1 (Estimate Power Load)	P_X
285	odb_pwr_si	Power load for each SI (watts; index: 1=ACIS-I, 2=ACIS-S, 3=HRC-I, 4=HRC-S)	Real	4	32	TBD	4.1.6.1 (Estimate Power Load)	P_x
286	odb_pwr_grating	Power load for switching out a grating (watts)	Real		8	TBD	4.1.6.1 (Estimate Power Load)	P_x
287	odb_pwr_sim	Power load during SIM repositioning (watts)	Real		8	TBD	4.1.6.1 (Estimate Power Load)	P_X
288	odb_pwr_transp	Power load for communications transmitter (watts; index=transmitter)	Real	2	16	TBD	4.1.6.1 (Estimate Power Load)	P_X
289	odb_pwr_lga	Power load for each LGA with power amplifier (watts; index=LGA)	Real	2	8	TRW EPS	4.1.6.1 (Estimate Power Load)	P_X
290	odb_pwr_momgmt	Power load for momentum management maneuvers (watts)	Real		8	TRW EPS	4.1.6.1 (Estimate Power Load)	P_X
291	odb_pwr_eclipse	Power load preparing for eclipse entry and recovery after eclipse exit (watts; index: 1=entry, 2=exit)	Real		8	TRW EPS	4.1.6.1 (Estimate Power Load)	P_X
291A	odb_lowpwr_alt	Defines the altitude below which the transmitters must be commanded to low power (kilometers)	Real		1	TRW EPS	4.4.6.3 Expand COMM or PBK Request	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
291B	odb_batrr	Defines the recharge ratio used to compute battery amp-hours-in	Real		8	MSFC	4.4.1.3 (Compute Battery State of Charge)	RR
291C	odb_battdelt	Defines the time between data scans for computing battery amp-hours-in, amp-hours-out, watt-hours-in, and watt-hours-out	Real		8	MSFC	4.4.1.2 (Compute Battery Watt-Hours-in and Watt-Hours-Out), 4.4.1.3 (Compute Battery State of Charge)	ΔT
	ODE_Earth_Characteristics		**Variable**		**80**			
292	odb_earth_grav_constant	The Earth gravitational constant in kilometers cubed per seconds squared.	Real		8	CSC	4.5.1.1 (Define AXAF Force Model)	μ
293	odb_earth_radius	The Earth's radius in kilometers.	Real		8	CSC	4.1.11, 4.3.1.1.6, 4.3.2.3.5, 4.5.1.1, 4.5.2.2.1, 4.5.2.2.2	R_E R_e
294	odb_earth_mass	The Earth's mass in kilograms.	Real		8	CSC	4.1.3.12.1 (Calculate S/C Momentum during maneuvers), 4.1.3.12.2, 4.5.3, 4.5.4.5	M_E
295	odb_em_mass_ratio	The earth-moon mass ratio.	Real		8	CSC	N/A[2]	
296	odb_espin	Spin rate of earth in deg/sec (corrected to include effect of precession)	Real		8	CSC	4.5.2.5, 4.5.4.18 (Calculate Ground Tracks, Calculate the Position of a DSN Station in GCI Coordinates)	ω_e

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
297	odb_esqrt_gm	Square root of the earth's gravitational constant	Real		8	CSC	4.5.3 (Estimate PCAD ephemeris Coefficients)	$\sqrt{\mu}$
298	odb_eflat	Earth oblateness correction factor	Real		8	CSC	N/A[2]	
299	odb_earies	Transit time of first point of aries (right ascension of Greenwich) (seconds from base reference time)	Real		8	CSC	4.5.4.9 (Compute Right Ascension of Greenwich Meridian)	t_{ref}
300	odb_oblecl	Defines the obliquity of the ecliptic on 12:00 UTC, January 1, 2000 (degrees)	Real		8	CSC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducual Lights)	N/A_1
301	odb_oerate	Defines the rate of change of the obliquity of the ecliptic (degrees/day)	Real		8	CSC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A_1
	ODE_Moon_Characteristics		**Variable**		**80**			
302	odb_moon_gm	The moon gravitational constant (kilometers[3] per seconds[2]).	Real		8	CSC	4.5.1.1 (Define AXAF Force Model)	μ_{moon}
303	odb_moon_radius	The radius of the moon (kilometers).	Real		8	CSC	4.1.4.2 (Check Moon Occultation)	ρ
304	odb_moon_ps	defines the lunar synodic period (seconds)	Real		8	CSC	4.5.2.2.3 (Determine Moon Shadow Entry/Exit Times)	$P_{synodic}$
305	odb_moon_mass	defines the mass of the moon (kilograms)	Real		8	CSC	4.5.2.2.3 (Determine Moon Shadow Entry/Exit Times)	M_M

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Sun_Characteristics		**Variable**		**80**			
306	odb_sun_gm	The Sun gravitational constant in kilometers^3 per seconds^2.	Real		8	CSC	4.5.1.1 (Define AXAF Force Model)	μ_{sun}
307	odb_sun_radius	The Sun radius in kilometers.	Real		8	CSC	4.5.2.2.1 (Estimate the Shadow Entry and Exit Time)	R_s
308	odb_sun_mom_flux	Mean momentum flux acting on a surface normal to the Sun's radiation (kg/m^2/s^2)	Real		8	CSC	4.1.3.12.1 (Calculate S/C Momentum during maneuvers) (Calculate Spacecraft Momentum During Maneuvers)	P
309	odb_sun_lon_rate	Defines the rate of change of the solar ecliptic longitude (degrees/second)	Real		8	CSC	4.1.3.12.n (Select Acquisition Stars, Guide Stars, and Fiducial Lights)	N/A[1]
	ODE_Math_Characteristics		**Variable**		**80**			
310	odb_rad_to_deg	The radians to degrees conversion factor.	Real		8	CSC	N/A	
311	odb_deg_to_rad	The degrees to radians conversion factor.	Real		8	CSC	N/A	
312	odb_pi	The value of pi.	Real		8	CSC	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
313	odb_speed_of_light	The speed of light in kilometers per second.	Real		8	CSC	4.3.1.1.8, 4.3.1.2.3.2, 4.4.2.1.2 (Construct Solar Reference Vectors, Determine Coarse and Fine Attitude, Extrapolate Present Trend Data)	c
314	odb_au	The astronomical unit (unit of length equal to the semi-major axis of the Earth's orbit about the Sun). (kilometers)	Real		8	CSC	N/A	
314A	odb_kgmss_to_flbs	Angular momentum in kilogram-meter^2/seconds to foot-lb force-seconds conversion factor	Real		8	CSC	N/A	
314B	odb_flbs_to_kgmss	Angular momentum in foot-lb force-seconds to kilogram-meter^2/seconds conversion factor	Real		8	CSC	N/A	
	ODE_Command_Characteristics		**Variable**		**80**			
315	odb_cmd_offset	Defines the offset between the start of the command data that the OFLS system generates and the command header which the ONLS system generates (bits)	Integer		4	CSC	N/A	
316	odb_nm_bits_word	Defines the number of bits in a word as used in MSFC-DOC-1949 to define the starting location of command fields	Integer		4	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
317	odb_uplink_rate	Defines the command load uplink rate (bits/second)	Integer		4	TRW SE/FSW	4.2.2.4 (Schedule Load Uplinks)	N/A
318	odb_abs_time_wait	Defines the command mnemonic for the absolute time delay command	ASCII		20	TRW SE/FSW	N/A	
319	odb_rel_time_wait	Defines the command mnemonic for the relative time delay command	ASCII		20	TRW SE/FSW	N/A	
320	odb_clear_obc_buf	Defines the command mnemonic for the clear memory load buffer command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
321	odb_complete_obc_buf	Defines the command mnemonic for the complete memory load buffer command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
322	odb_obc_buf_chksum	Defines the command mnemonic for the OBC checksum command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
323	odb_call_scs	Defines the command mnemonic for the call SCS command	ASCII		20	TRW SE/FSW	N/A	
324	odb_scs_buf_header	Defines the command mnemonic for the SCS load buffer header command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
325	odb_mem_ld_header	Defines the command mnemonic for the absolute memory load command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
325A	odb_mem_hdr_xs_fld	Defines the field mnemonic for the OBC processor load checksum bypass bit field	ASCII		20	TRW SE/FSW	NA	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
326	odb_scs_blk_header	Defines the command mnemonic for the SCS load block header command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
327	odb_mem_blk_header	Defines the command mnemonic for the absolute memory load block command; index = OBC processor	ASCII	2	40	TRW OPS	N/A	
328	odb_mem_ld_cmd	Defines the command mnemonic for the absolute memory load command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
329	odb_end_scs	Defines the command mnemonic for the end SCS command	ASCII		20	TRW SE/FSW	N/A	
330	odb_scs_ld_cmd	Defines the command mnemonic for the SCS load command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
331	Reserved							Deleted ECP
332	odb_cpe_mem_ld	Defines the command mnemonic for the CPE RAM load command; index = CPE processor	ASCII	2	40	TRW SE/FSW	N/A	
332A	odb_cpemem_xs_fld	Defines the field mnemonic for the CPE processor load checksum bypass bit field	ASCII		20	TRW SE/FSW	NA	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
333	odb_iu_swld_hder	Defines the command mnemonic for the IU EEPROM software load header command; index = IU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
334	odb_iu_swld_data	Defines the command mnemonic for the IU EEPROM software load data command; index = IU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
335	odb_iu_swld_trlr	Defines the command mnemonic for the IU EEPROM software load trailer command; index = IU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
336	odb_ctu_swld_hder	Defines the command mnemonic for the CTU EEPROM software load header command; index = CTU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
337	odb_ctu_swld_data	Defines the command mnemonic for the CTU EEPROM software load data command; index = CTU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
338	odb_ctu_swld_trlr	Defines the command mnemonic for the CTU EEPROM software load trailer command; index = CTU EEPROM processor	ASCII	2	40	TRW SE/FSW	N/A	
339	odb_cpe_buf_header	Defines the command mnemonic for the CPE RAM load buffer header command; index = CPE processor	ASCII	2	40	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
340	odb_cpe_complete_buf	Defines the command mnemonic for the CPE RAM load complete load command; index = CPE processor	ASCII	2	40	TRW SE/FSW	N/A	
341	odb_cpe_clear_buf	Defines the command mnemonic for the CPE RAM load clear buffer command; index = CPE processor	ASCII	2	40	TRW SE/FSW	N/A	
342	odb_cpe_ld_chksum	Defines the command mnemonic for the CPE RAM checksum command; index = CPE processor	ASCII	2	40	TRW SE/FSW	N/A	
343	odb_aca_buf_header	Defines the command mnemonic for the ACA processor load buffer header command; index = ACA processor	ASCII	2	40	TRW SE/FSW	N/A	
344	odb_aca_buf_data	Defines the command mnemonic for the ACA processor load data command; index = ACA processor	ASCII	2	40	TRW SE/FSW	N/A	
345	odb_aca_buf_chksum	Defines the command mnemonic for the ACA processor load checksum command; index = ACA processor	ASCII	2	40	TRW SE/FSW	N/A	
346	odb_sim_swld_hdr	Defines the command mnemonic for the SIM software load header command; index = SIM processor	ASCII	2	40	TRW SE/FSW	N/A	
347	odb_sim_swld_data	Defines the command mnemonic for the SIM software load data command; index = SIM processor	ASCII	2	40	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
348	odb_sim_swld_trlr	Defines the command mnemonic for the SIM software load trailer command; index = SIM processor	ASCII	2	40	TRW SE/FSW	N/A	
349	odb_eio_swld_hdr	Defines the command mnemonic for the EIO software load header command; index = EIO processor	ASCII	2	40	TRW SE/FSW	N/A	
350	odb_eio_swld_data	Defines the command mnemonic for the EIO software load data command; index = EIO processor	ASCII	2	40	TRW SE/FSW	N/A	
351	odb_eio_swld_trlr	Defines the command mnemonic for the EIO software load trailer command; index = EIO processor	ASCII	2	40	TRW SE/FSW	N/A	
352	odb_simpkt_hdr	Defines the command mnemonic for the SIM packet header command	ASCII		20	TRW SE/FSW	N/A	
353	odb_acapkt_hdr	Defines the command mnemonic for the ACA packet header command	ASCII		20	TRW SE/FSW	N/A	
354	odb_eiopkt_hdr	Defines the command mnemonic for the EIO packet header command	ASCII		20	TRW SE/FSW	N/A	
355	odb_ld_acis_cmd	Defines the command mnemonic for the ACIS load command	ASCII			TRW SE/FSW	N/A	
356	odb_ld_rt_cmd	Defines the command mnemonic for the real-time load command; index = OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
357	odb_end_scs	Defines the command mnemonic for the end SCS command	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
358	odb_ld_rt_data_fd	Defines the field mnemonic for the data field in the real time load data command	ASCII		20	TRW SE/FSW	N/A	
359	odb_scs_ld_data_fd	Defines the field mnemonic for the data field in the SCS load command	ASCII		20	TRW SE/FSW	N/A	
360	odb_mem_ld_data_fd	Defines the field mnemonic for the data field in the absolute memory load command	ASCII		20	TRW SE/FSW	N/A	
361	odb_cpe_ld_data_fd	Defines the field mnemonic for the data field in the CPE RAM load data command	ASCII		20	TRW SE/FSW	N/A	
362	odb_iu_ld_data_fd	Defines the field mnemonic for the data field in the IU EEPROM load data command	ASCII		20	TRW SE/FSW	N/A	
363	odb_ctu_ld_data_fd	Defines the field mnemonic for the data field in the CPE EEPROM load data command	ASCII		20	TRW SE/FSW	N/A	
364	odb_aca_ld_data_fd	Defines the field mnemonic for the data field in the ACA processor load data command	ASCII		20	TRW SE/FSW	N/A	
365	odb_scs_hdr_blk_ct	Defines the field mnemonic for the block count field in the SCS load buffer header command	ASCII		20	TRW SE/FSW	N/A	
366	odb_scs_hdr_buff_ct	Defines the field mnemonic for the buffer command count field in the SCS load buffer header command	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
367	odb_scs_hdr_step_ct	Defines the field mnemonic for the SCS step counter data field in the SCS load block header command	ASCII		20	TRW SE/FSW	N/A	
368	odb_scs_hdr_id	Defines the field mnemonic for the SCS id field in the SCS load block header command	ASCII		20	TRW SE/FSW	N/A	
369	odb_scs_hder_cmd_ct	Defines the field mnemonic for the SCS command count field in the SCS load block header command	ASCII		20	TRW SE/FSW	N/A	
370	odb_mem_hdr_blk_ct	Defines the field mnemonic for the block count field in the absolute memory load buffer header command	ASCII		20	TRW SE/FSW	N/A	
371	odb_mem_hdr_buff_ct	Defines the field mnemonic for the buffer command count field in the absolute memory load buffer header command	ASCII		20	TRW SE/FSW	N/A	
372	odb_mem_hdr_cmd_ct	Defines the field mnemonic for the 16-bit word count field in the absolute memory load block header command	ASCII		20	TRW SE/FSW	N/A	
373	odb_mem_hdr_msb	Defines the field mnemonic for the MSB portion of the physical address field in the absolute memory load block header command	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
373A	odb_mem_hdr_lsb	Defines the field mnemonic for the LSB portion of the physical address field in the absolute memory load block header command	ASCII		20	TRW SE/FSW	N/A	
374	odb_rel_tw_fld	Defines the field mnemonic for the relative time delay command minor cycle counts data field	ASCII		20	TRW SE/FSW	N/A	
375	odb_ab_tw_fld	Defines the field mnemonic for the absolute time delay command VCDU counts data field	ASCII		20	TRW SE/FSW	N/A	
376	odb_call_scs_fld	Defines the field mnemonic for the call SCS command SCS ID data field	ASCII		20	TRW SE/FSW	N/A	
377	odb_simpkt_chksm_fld	Defines the field mnemonic for the SIM packet checksum data field	ASCII		20	TRW SE/FSW	N/A	
378	odb_acapkt_chksm_fld	Defines the field mnemonic for the ACA packet checksum data field	ASCII		20	TRW SE/FSW	N/A	
379	odb_eiopkt_chksm_fld	Defines the field mnemonic for the EIO packet checksum data field	ASCII		20	TRW SE/FSW	N/A	
380	odb_simpkt_cmd_ct	Defines the field mnemonic for the SIM packet command count data field	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
381	odb_acapkt_cmd_ct	Defines the field mnemonic for the ACA packet command count data field	ASCII		20	TRW SE/FSW	N/A	
382	odb_eiopkt_cmd_ct	Defines the field mnemonic for the EIO packet command count data field	ASCII		20	TRW SE/FSW	N/A	
383	odb_cpe_addr_fld	Defines the field mnemonic for the CPE starting address data field	ASCII		20	TRW SE/FSW	N/A	
384	odb_iu_addr_msb_fld	Defines the field mnemonic for the IU EEPROM starting address MSB data field	ASCII		20	TRW SE/FSW	N/A	
384A	odb_iu_addr_lsb_fld	Defines the field mnemonic for the IU EEPROM starting address LSB data field	ASCII		20	TRW SE/FSW	N/A	
385	odb_ctu_addr_msb_fld	Defines the field mnemonic for the CTU EEPROM starting address MSB data field	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
385A	odb_ctu_addr_lsb_fld	Defines the field mnemonic for the CTU EEPROM starting address LSB data field	ASCII		20	TRW SE/FSW	N/A	
386	odb_sim_addr_fld	Defines the field mnemonic for the SIM starting address data field	ASCII		20	TRW SE/FSW	N/A	
387	odb_aca_addr_fld	Defines the field mnemonic for the ACA starting address data field	ASCII		20	TRW SE/FSW	N/A	
388	odb_eio_addr_fld	Defines the field mnemonic for the EIO starting address data field	ASCII		20	TRW SE/FSW	N/A	
389	odb_upl_ephem_coeff	Defines the command sequence mnemonic for the ephemeris coefficient update command	ASCII		20	TRW SE/FSW	4.5.3 (Estimate PCAD Ephemeris Coefficients)	N/A
390	odb_upl_gyro_bias	Defines the command sequence mnemonic for the gyro bias update command	ASCII		20	TRW SE/FSW	4.3.1.7 (PCAD Uplink Parameters)	N/A
391	odb_upl_gyro_misal	Defines the command sequence mnemonic for the gyro scale factor misalignment update command	ASCII		20	TRW SE/FSW	4.3.1.7 (PCAD Uplink Parameters)	N/A
392	odb_upl_fss_misal	Defines the command sequence mnemonic for the FSS to ACA misalignment update command	ASCII		20	TRW SE/FSW	4.3.1.7 (PCAD Uplink Parameters)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
393	odb_upl_att_cor	Defines the command sequence mnemonic for the attitude correction update command	ASCII		20	TRW SE/FSW	4.3.1.7 (PCAD Uplink Parameters)	N/A
394	odb_obc_hwld_hder	Defines the command mnemonics for the OBC hardware load header commands; index: 1 = OBC processor, 2 = command or multipart command mnemonic	ASCII	2x80	3200	TRW SE/FSW	N/A	
395	odb_obc_hwld_data	Defines the command mnemonics for the OBC hardware load header commands; index: OBC processor	ASCII	2	40	TRW SE/FSW	N/A	
396	odb_obc_hwld_trlr	Defines the command mnemonics for the OBC hardware load trailer commands; index: 1 = OBC processor, 2 = command or multipart command mnemonic	ASCII	2x20	800	TRW SE/FSW	N/A	
397	Deleted							
398	odb_obchwld_addr_msb_fd	Defines the field mnemonic for the OBC hardware load address MSB field; index: 1 = mnemonic (1 = mnemonic for the command that includes the starting address field, 2 = field mnemonic), 2 = OBC processor (1=A, 2=B)	ASCII	2x2	80	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
398A	odb_obchwld_saddr_lsb_fd	Defines the field mnemonic for the OBC hardware load starting address LSB field; index: 1 = mnemonic (1 = mnemonic for the command that includes the starting address field, 2 = field mnemonic), 2 = OBC processor (1=A, 2=B)	ASCII	2x2	80	TRW SE/FSW	N/A	
399	odb_obchwld_eaddr_lsb_fd	Defines the field mnemonic for the OBC hardware load ending address LSB field; index: 1 = mnemonic (1 = mnemonic for the command that includes the starting address field, 2 = field mnemonic), 2 = OBC processor (1=A, 2=B)	ASCII	2x2	80	TRW SE/FSW	N/A	
400	odb_obchwld_chksum_fd	Defines the field mnemonic for the OBC hardware load checksum field; index: 1 = mnemonic (1 = mnemonic for the command that includes the starting address field, 2 = field mnemonic), 2 = OBC processor (1=A, 2=B)	ASCII	2x2	80	TRW SE/FSW	N/A	
400A	odb_obcsw_cmd	Defines integer values for ground bits 12 through 16 used to identify commands to the OBC software; index = value	Integer	2	8	TRW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
401	odb_rthw_wrap	Defines the command mnemonic for the real-time hardware commands in a command load data file; index: hardware bus (1=bus A, 2=bus B)	ASCII	2	40	TRW SE/FSW	N/A	
402	odb_obchw_addr_fd	Defines the field mnemonic for the OBC hardware-direct memory load header command start address field	ASCII		20	TRW SE/FSW	N/A	
403	odb_obchw_len_fd	Defines the field mnemonic for the OBC hardware-direct memory load header command load length field	ASCII		20	TRW SE/FSW	N/A	
404	odb_obchw_data_fd	Defines the field mnemonic for the OBC hardware-direct memory load data command data field	ASCII		20	TRW SE/FSW	N/A	
405	odb_obchw_chksm_fd	Defines the field mnemonic for the OBC hardware-direct memory load trailer command checksum field	ASCII		20	TRW SE/FSW	N/A	
406	odb_cpe_hwld_hder	Defines the command mnemonic for the CPE hardware-direct memory load header command; index: hardware bus (1=CPE A, 2=CPE B)	ASCII	2	40	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
407	odb_cpe_hwld_data	Defines the command mnemonic for the CPE hardware-direct memory load data command; index: hardware bus (1=CPE A, 2=CPE B)	ASCII	2	40	TRW SE/FSW	N/A	
408	odb_cpe_hwld_trlr	Defines the command mnemonic for the CPE hardware-direct memory load trailer command; index: hardware bus (1=CPE A, 2=CPE B)	ASCII	2	0	TRW SE/FSW	N/A	
409	odb_cpehw_addrmsb__fd	Defines the field mnemonic for the CPE hardware-direct memory load header command start address MSB field	ASCII		20	TRW SE/FSW	N/A	
409A	odb_cpehw_addrlsb__fd	Defines the field mnemonic for the CPE hardware-direct memory load header command start address LSB field	ASCII		20	TRW SE/FSW	N/A	
410	odb_cpehw_len_fd	Defines the field mnemonic for the CPE hardware-direct memory load header command load length field	ASCII		20	TRW SE/FSW	N/A	
411	odb_cpehw_data_fd	Defines the field mnemonic for the CPE hardware-direct memory load data command data field	ASCII		20	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
412	odb_cpehw_chksm_fd	Defines the field mnemonic for the CPE hardware-direct memory load trailer command checksum field	ASCII		20	TRW SE/FSW	N/A	
413	odb_st_acis_pkt_cmd	Defines the command mnemonic for the OBC command used for the stored ACIS packet load data	ASCII		20	TRW SE/FSW	N/A	
414	odb_st_acis_pkt_data_fld	Defines the field mnemonic for the OBC command used for the data field in the stored ACIS packet load data command	ASCII		20	TRW SE/FSW	N/A	
415	odb_rt_acis_pkt_cmd	Defines the command mnemonic for the OBC command used for the real-time ACIS packet load data	ASCII		20	TRW SE/FSW	N/A	
416	odb_st_acis_pkt_data_fld	Defines the field mnemonic for the OBC command used for the data field in the real-time ACIS packet load data command	ASCII		20	TRW SE/FSW	N/A	
417	odb_obc_cmd_wrap	Defines the command mnemonic for the real-time OBC command wrapper used for building load data files; index: hardware bus (1=OBC A, 2=OBC B)	ASCII	2	40	TRW SE/FSW	N/A	
418	odb_obc_mem_wrap	Defines the command mnemonic for the OBC memory load data command wrapper used for building load data files; index: hardware bus (1=OBC A, 2=OBC B)	ASCII	2	40	TRW SE/FSW	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
419	odb_noop_wrap	Defines the command mnemonic for the real-time OBC NOOP command wrapper used for building load data files; index: hardware bus (1=OBC A, 2=OBC B)	ASCII	2	40	TRW SE/FSW	N/A	
420	odb_obcfnc_cmd	Defines integer values for ground bits 20 through 23 used to identify commands to specific OBC software functions; index = software function (1 = Executive, 2 = CCDM, 3 = Health and Status, 4 = PCAD, 5 = EPS, 6 = Telescope Support)	Integer	6	24	TRW FSW	N/A	
421	odb_ccdm_cmd	Defines integer values for ground bits 16 through 19 used to identify commands to the OBC CCDM function; index = value	Integer	2	8	TRW FSW	N/A	
422	odb_mpdata_cmd	Defines the integer values for ground bits 16 through 19 used to identify the data portion of a multipart command	Integer		8	TRW FSW	N/A	
423	odb_ccdm2_cmd	Defines integer values for ground bits 12 through 16 used to identify commands to the OBC CCDM functions; index = value	Integer	11	8	TRW FSW	N/A	
423a	odb_trnoop_cmd	Defines the mnemonic for CTU NOOP commands; used to ensure sufficient bit transition density	ASCII		20	TRW CCDM	N/A	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length / Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Object_Characteristics		**Variable**		**80**			
424	odb_num_targets	Number of solar system object (excluding Earth, Moon, Sun and planets) ephemerides defined in odb_target_ephem.	Integer		4	ASC	4.1.3.5 Position target on Optical Axis	N/A
425	odb_target_ephem	Solar system object ephemeris for use in target scheduling; index: 1 = target ephemeris (1 = semi-major axis (kilometers), 2 = eccentricity, 3 = inclination (degrees), 4 = right ascension of the ascending node (degrees) , 5 = argument of perigee (degrees), 6 = mean anomaly (degrees)), 2 = solar system objects	Real	6x10	480	ASC	4.1.3.5 Position target on Optical Axis	N/A
426	odb_target_epoch	Epoch time of each solar system object (excluding Earth, Moon, Sun, and planets) for use in target scheduling (HOSC GMT); index = solar system object	ASCII	10	210	ASC	4.1.3.5 Position target on Optical Axis	N/A
427	odb_target_name	Names of each solar system object (excluding earth, Moon, Sun, and planets) for use in target scheduling (HOSC GMT); index = solar system object	ASCII	10	200	ASC	4.1.3.5 Position target on Optical Axis	N/A

3.3 COMMAND DEFINITION TABLES

3.3.1 HEADER INFORMATION

Element Type:	CDB
Format Provided By:	MOL
Data Provided By:	TRW
Data Routinely Accessed By:	ONLS (DBCR, CMD Update Application), OFLS (CM)
Format structure:	Relational Database Management System (RDBMS) table format

Delivery Media:

Initially the ASCII text files will be file transferred from a workstation in Redondo Beach to a workstation at the OCC.

3.3.2 CONTENT

The Command Definition Tables are used by the Enhanced HOSC System (EHS) to populate the Operational Command Database (OCDB). The OCDB is then used to configure the EHS command processing system. The OFLS uses the OCDB to build ATS and RTS command load files and planned real-time commands to be uplinked by the ONLS to the spacecraft. The information provided in the Command Definition Tables includes command owners, definition of command headers and fields, definition of commands and fields, decalibration information, and telemetry verifiers. TRW will provide the Command Definition Tables to be used for supporting the AXAF Project.

3.3.3 FORMAT DESCRIPTION

When the Command Definition Tables are loaded onto the EHS they will be in a Relational Database Management System (RDBMS) table format. These relational tables are defined in the MSFC Huntsville Operations Support Center Command Database Definition (MSFC-DOC-1949 Volume 2). However, TRW will provide the tables in an ASCII text file format which is also defined in the MSFC-DOC-1949 Volume 2. In this format there will be an ASCII text file for each one of the relational tables. For a detailed definition see the MSFC-DOC-1949 Volume 2.

3.4 COMMAND DEFINITION TABLES, MULTI-PART

3.4.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	TRW
Data Routinely Accessed By:	OFLS (CM)

Format structure: ASCII Flat File

Delivery Media: Electronic Transfer

3.4.2 CONTENT

The Multi-Part Command Definition Tables are used by the OFLS to build ATS and RTS command load files and planned real-time commands to be uplinked by the ONLS to the spacecraft. The information provided in the Multi-Part Command Definition Tables includes the definition of pseudo-headers, pseudo-trailers, commands and fields, decalibration information, telemetry verifiers and special processing information for all multi-part commands defined for AXAF. The Multi-Part Command Definition Tables only define the formats for the 28 bits of command data provided by the OFLS to the ONLS. TRW will provide the Multi-Part Command Definition Tables used for supporting the AXAF project.

The Multi-Part Command Definition Tables are specified in 8 text files. The directory file must have the name MPLIST.TXT and contains the list of names for the pseudo-header, pseudo-trailer, command, command field, decalibration, telemetry verifier, and special processing tables. The names of the files stored in the ODE must match the names of the files specified in the MPLIST.TXT file.

3.4.3 FORMAT DESCRIPTION

The format for Multi-Part Command Definition Tables is nearly identical to the Command Definition Tables as defined in section 3.3 of this document. In this format there will be an ASCII text file for each one of the relational tables. The Pseudo-Header Table and the Pseudo-Header Field table define header-like fields that repeat at the start of the multi-part commands. They do not include the headers and header fields that contain command routing information that the ONLS system adds to the command data generated by the OFLS. The Pseudo-Header Table (Table 3-2), Pseudo-Header Field table (Table 3-3), Command Table (Table 3-4) and the Command Field Table (Table 3-5) contain data not defined in Command Definition Tables and also do not use all of the fields defined for each Command Definition Table. In order to retain correspondence to the formats defined in the Command Definition Tables, fields that are not used have been retained and labeled “spare” and new fields have been appended to the end of the record structure. The Point Pair De-calibration Table (Table 3-6) and the Telemetry Verifier Table (Table 3-7) definitions are identical to their definitions in the Command Definition Tables. The Special Processing Table (Table 3-8) is a table created just for the Multi-part Command Definitions.

This table defines the special processing for the parity field and the sequence counter field. The Pseudo-Trailer Table (Table 3-9) and the Pseudo-trailer Field Table (Table 3-10) define tables created just for the Multi-part Command Definitions. These tables define header-like fields that repeat for each 28 bit segment of a multi-part command. However, pseudo-headers repeat at the start of each 28 bit segment of a multi-part command, while pseudo-trailers repeat at the end of each 28 bit segment.

The OFLS builds multipart commands by formatting the pseudo-header in the starting bits of a 28 bit field as defined by the Pseudo-Header tables and the pseudo-trailer in the ending bits of a 28 bit field as defined by the Pseudo-Trailer tables. The data portion of the

command is built as defined by the Command tables. The resulting bit stream is then segmented according to the number of bits in the 28 bit field between the end of the pseudo-header and the start and the pseudo-trailer. The entire 28 bit segment (pseudo-header, n bits of the command data, pseudo-trailer) are repeated until all of the command data has been built into 28 bit segments. If the command data is not an exact multiple of the n bits available in the 28 bit format, the last command is zero filled.

The following is a list of formatting rules required in addition to the validation rules defined for the Command Definition Tables:

- data values cannot contain commas (,) or semicolons (;).
- white space can only be contained in the description field of a table.

These rules replace rule number 5 in section 3.2 of the MSFC-DOC-1949 Vol. II.

Table 3-2 Pseudo Header Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
PSEUDO_HEADER_ID	20	ASCII	Unique ID assigned to a command pseudo header
SPARE	20	ASCII	Unused
PSEUDO_TRAILER_ID	20	ASCII	Unique ID assigned to a command pseudo-trailer
SPARE	1	ASCII	Unused
SPARE	5	ASCII	Unused
LENGTH	3	integer	Indicates the total length of the pseudo header in bits.
SPARE	100	ASCII	Unused
SEQ_COUNT_TYPE	3	ASCII	Indicates the sequence counter type for the pseudo header. S3R - 3 bit long sequence counter that begins at 0 stops at 7 and rolls over back to 0 S3H - 4 bit long sequence counter that begins at 1 and stops and holds at 3

Table 3-3 Pseudo Header Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
PSEUDO_HEADER_ID	20	ASCII	Unique ID assigned to a multi-part command pseudo-header
FIELD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic assigned to a pseudo-header field in a command

Table 3-3 Pseudo Header Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
			pseudo-header.
FIELD_TYPE	1	ASCII	Indicates whether this field is predefined or modifiable. “P” - Predefined field. “M” - Modifiable field.
INPUT_DATA_TYPE	1	ASCII	Indicates the type of input data.
UPLINK_DATA_TYPE	5	ASCII	Indicates how the data will be converted before uplink. The following are the uplink data types supported by the OFLS: IDIS - Binary integer discrete IMAG - Signed integer ITWO - Two’s complement signed integer IUNS - Unsigned integer FMIL - MIL-STD-1750A floating point IPAR - Parity bit field ISEQ - Sequence counter field
LENGTH	3	integer	Indicates the total length of the field in bits.
START_WORD	3	integer	The word within the pseudo-header where the field begins. The first word of the pseudo-header is word 1.
START_BIT	2	integer	The bit within the start word where the field begins. The first bit of a word is bit 0.
INIT_DATA	32	ASCII	Actual data to be located in the field specified by this record. Format must be consistent with the INPUT_DATA_TYPE defined for the field. NOTE: Field length is 32 characters for consistency with 1949. However, pseudo-header field definitions can never be more than 7 characters (28 bits).
SPARE	1	ASCII	Unused
SPARE	100	ASCII	Unused

Table 3-4 Command Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
CMD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic assigned to a command.
MSID	20	ASCII	Wire destination identifier assigned to each command. No special characters allowed.
OWNER_ID	20	ASCII	Unique identifier that indicates the responsible and authorized owner of this command.
SPARE	1	ASCII	Unused
START_ADDRESS	5	integer	Memory address identifying the starting location of the parameter (hexadecimal). Required for OBC/CPE Tables format only
CLASS	10	ASCII	Class to which a command belongs.
SPARE	3	ASCII	Unused
LENGTH	5	integer	Total length in words of all the fields for a given command without the header and the pseudo header fields. Maximum length of the command for variable length commands.
VAR_LENGTH	1	ASCII	Indicates whether this command is variable length. “Y” - variable length command. “N” - fixed length command.
SPARE	1	ASCII	Unused
SPARE	1	ASCII	Unused
SPARE	1	ASCII	Unused
CRITICAL	1	ASCII	Indicates whether this command is critical or not. “Y”- critical command. “N” - non-critical command.
PSEUDO_TRAILER_ID	20	ASCII	Unique ID assigned to a command pseudo-trailer
SPECIAL_PROCESS	1	ASCII	Indicates the type of special processing for this command.

Table 3-4 Command Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
			"I" - IOE command "D" - Dense multi-part command "S" - Sparse multi-part command
PARITY_PROCESS	1	ASCII	Indicates the type of parity calculation for this command. "1" - Real time command parity calculation "R" - IOE command parity calculation "Q" - CPE command parity calculation
TIME_CONSTRAINT	14	real	Number of seconds that must elapse after issuing this command before another command can be issued. Includes decimal point, if applicable.
PSEUDO_HEADER_ID	20	ASCII	Identifier for pseudo-header to be used for this command.
TLM_VERIF_FLAG	1	ASCII	Indicates whether or not this command has telemetry verifiers associated with it. "Y" - Yes. "N" - No.
SPARE	14	ASCII	Unused
SPARE	89	ASCII	Unused
DESCRIPTION	250	ASCII	Text description.

Table 3-5 Command Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
CMD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic assigned to the command to which the field belongs.
FIELD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic for the command field.
FIELD_TYPE	1	ASCII	Indicates whether this field is predefined or modifiable. "P" - Predefined field. "M" - Modifiable field.

Table 3-5 Command Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
INPUT_DATA_TYPE	1	ASCII	Indicates the type of input data.
UPLINK_DATA_TYPE	5	ASCII	Indicates how the data will be converted before uplink. The following are the uplink data types supported by the OFLS: IDIS - Binary integer discrete IMAG - Signed integer ITWO - Two's complement signed integer IUNS - Unsigned integer FMIL - MIL-STD-1750A floating point IPAR - Parity bit field INWD - Number of data words field ICP - Checksum calculation for the OFP hardware (OBC Buffers) ICF - Checksum calculation for the OFF hardware (CPE) ICA - Checksum calculation for the ACA, EPHIN and the SIM packet commands ICEP – Checksum calculation for the I-EPHIN processor load ICSP – Checksum calculation for the SIM processor load IMx – MSB portion of input data value where x = O, indicates an OBC address x = C, indicates a CPE address x = I, indicates an IU EEPROM address x = T, indicates a CTU EEPROM address ILx – LSB portion of input data value where x = O, indicates an OBC address x = C, indicates a CPE address x = I, indicates an IU EEPROM address x = T, indicates a CTU EEPROM address
ENG_UNIT	10	ASCII	Indicates the engineering units associated with the field.
DIMENSION	30	ASCII	Indicates the physical property associated with the engineering units.

Table 3-5 Command Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
DECAL_TYPE	2	ASCII	Indicates the type of decalibration associated with a command. "N" - No decalibration. "PC" - Polynomial Coefficient decalibration. "PP" - Point Pair decalibration.
LENGTH	5	integer	Indicates the total length of the field in bits. For MSB/LSB processing (UPLINK_DATA_TYPE = IMx or ILx), this field also indicates the number of MSBs or LSBs to be extracted from the input data value.
START_WORD	3	integer	The word within the command where the field begins. The first word of the pseudo-header is word 1. If the command has no pseudo-header, then the first word of the command definition in the multipart command database is word 1.
START_BIT	2	integer	The bit within the start word where the field begins. The first bit of a word is bit 0.
INIT_DATA	32	ASCII	Actual command data to be located in the field specified by this record. Format must be consistent with the INPUT_DATA_TYPE defined for the field.
RANGE_LOW	16	real	Low end of range in Engineering Units for valid command field value. Includes sign and decimal point, if applicable.
RANGE_HIGH	16	real	High end of range in Engineering Units for valid command field value. Includes sign and decimal point, if applicable.
CAL_COEF_0	16	real	Calibration Coefficient 0. Used for polynomial decalibration.
CAL_COEF_1	16	real	Calibration Coefficient 1. Used for polynomial decalibration.
SPARE	100	ASCII	Unused
REPEAT_COUNT	4	integer	The maximum number of times the field can repeat.
PERIOD	4	integer	The period is used to calculate the starting location (bits) for each repetition

Table 3-5 Command Field Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
			of a repeatable field. The following formula indicates how to calculate the starting location for each repetition: rep_field_start$_n$ = field_offset+((n-1)*PERIOD) where rep_field_start$_n$ is the starting location of the data in bits, n is the nth occurrence of the repeatable field, and field_offset (bits) is the field offset calculated from the field START_WORD and field START_BIT.

Table 3-6 Point Pair Decalibration Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
CMD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic for the command to which field referenced by FIELD_MNEMONIC belongs.
FIELD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic of a command field to which point pair applies.
COUNTS	12	integer	Decimal integer representing the raw count value of the point pair.
VALUE	14	real	Engineering unit equivalent of the COUNTS column. Includes sign and decimal point, if applicable.

Table 3-7 Telemetry Verifier Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
CMD_MNEMONIC	20	ASCII	Unique user-friendly mnemonic for the command to which telemetry verifier table belongs.
TLM_MSID	20	ASCII	Unique identifier assigned to the telemetry measurement to be checked for a command's verification. No special character allowed.
STATE_CODE	12	ASCII	State conversion code of the measurement identified by TLM_MSID which indicates

Table 3-7 Telemetry Verifier Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
			positive command verification.
RANGE_LOW	14	real	Low end of range in Engineering Units for the measurement identified by TLM_MSID. If measurement value between this value and RANGE_HIGH, the verification is positive. Includes sign and decimal point, if applicable.
RANGE_HIGH	14	real	High end of range in Engineering Units for the measurement identified by TLM_MSID. Includes sign and decimal point, if applicable.

Table 3-8 Special Processing Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
SPEC_PROC_ID	3	ASCII	Special processing id that ties the special processing table to a command, psuedo-header, or command field referenced by the PARITY_PROCESS in the Command Table, the SEQ_COUNT_TYPE in the Pseudo Header Table, or the UPLINK_DATA_TYPE in the Command Field Table
START	4	integer	For parity bit processing: identifies the first bit of the command to be used in computing the parity. The first bit of the pseudo-header is bit 0. If the command has no pseudo-header, then the first bit of the command definition in the multipart command database is bit 0. For sequence counter processing: specifies the starting value for the sequence counter. Spare (not used) for MSB/LSB processing.
END	4	integer	For parity bit processing: identifies the last bit of the command to be used in computing the parity.. The first bit of the pseudo-header is bit 0. If the command has no pseudo-header, then the first bit of the command definition in the multipart command database is bit 0. For sequence counter processing: specifies the ending (hold or rollover) value

Table 3-8 Special Processing Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
			for the sequence counter. For MSB/LSB processing, indicates the total length of the original input data field from which the MSBs or LSBs are to be extracted. For example, if 12 MSBs are to be extracted from a 20 bit address, then the value of the length parameter in the command field table is 12, and the value of this parameter is 20
TYPE	1	ASCII	Identifies the special processing type. E - Even parity O - Odd parity R - Rollover N - No rollover B - MSB/LSB
VALUE	4	integer	The initial value of parity for parity special processing. Spare (not used) for sequence counter and MSB/LSB processing.

Table 3-9 Psuedo-Trailer Table Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
PSEUDO_TRAILER_ID	20	ASCII	Unique ID assigned to a command pseudo-trailer
SPARE	20	ASCII	Unused
SPARE	1	ASCII	Unused
SPARE	5	ASCII	Unused
LENGTH	3	Integer	Indicates the total length of the pseudo-trailer in bits. Allowed values are 1-27 bits
SPARE	100	ASCII	Unused
SPARE	3	ASCII	Unused

Table 3-10 Psuedo-Trailer Field Definition

COLUMN NAME	*MAXIMUM COLUMN LENGTH*	*COLUMN TYPE*	*DESCRIPTION*
PSEUDO_TRAILER_ID	20	ASCII	Unique ID assigned to a command pseudo-trailer.
FIELD_MENMONIC	20	ASCII	Unique user-friendly mnemonic assigned to a pseudo-trailer field in a command pseudo-trailer.
FIELD_TYPE	1	ASCII	Indicates whether this field is pre-defined or modifiable “P” – Predefined “M” – Modifiable.
INPUT_DATA_TYPE	1	ASCII	Indicates the type of input data.
UPLINK_DATA_TYPE	5	ASCII	Indicates how the data will be converted before uplink. The following are uplink data types supported by the OFLS: IDIS – Binary integer discrete IMAG – Signed integer ITWO – Two’s complement signed integer IUNS – Unsigned integer
LENGTH	3	Integer	Indicates the total length of the field in bits.
START_WORD	3	Integer	The word within the command where the field begins. The first word of a pseudo-trailer is 1. If the command has no pseudo-header, then the first word of the command definition in the multipart command database in word 1.
START_BIT	3	Integer	The bit within the start word where the field begins. The first bit of a word is bit 0.
INIT_DATA	32	ASCII	Actual data to be located in the field specified by this record. Format must be consistent with the INPUT_DATA_TYPE defined for the field. NOTE: Field length is 32 characters for consistency with 1949. However, pseudo-trailer field definitions can never be more than 7 characters (HEX) or 28 character (Binary).
SPARE	1	ASCII	Unused
SPARE	100	ASCII	Unused

3.5 COMMAND LOAD

3.5.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	OFLS (CM)
Data Routinely Accessed By:	ONLS, OFLS (CM)
Format Structure:	Sequential
Delivery Media:	Electronic Transfer

3.5.2 CONTENT

The Command Load data element contains command, data, or software load files to be uplinked to the spacecraft. A load file may contain an absolute time sequence (ATS) command load; a relative time sequence (RTS) command load; an OBC, ACIS, CPE, or AC table; or an OBC, CPE, ACIS, AC, or interface unit (IU) EEPROM flight software update.

Each Command Load file is composed of a header record containing information about the load followed by the data records composed of 28 bits of spacecraft command data.

3.5.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 200,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Command Load Header Record

Record Format: Fixed

Record Length: 65 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
comment_id	Field indicating the record is a comment. Value will be the pound sign, "#".	ASCII		1
cl_load_id	Load name which, along with the three character file extension, uniquely identifies the load	ASCII		12

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cl_load_type	Identifier for the type of data contained in the load. Allowed values are: 'ATS-A'=ATS stored command sequence load to OBC A; 'ATS-B'=ATS stored command sequence load to OBC B; 'RTS-A'=RTS stored command sequence load to OBC A; 'RTS-B'=RTS stored command sequence load to OBC B; 'OBC SW - A' = absolute memory flight software load to OBC A 'OBC SW - B' = absolute memory flight software load to OBC B 'CPE SW - B' = absolute memory flight software load to CPE B OBC SW/DL - A' = dead load (hardware format) flight software load to OBC A 'OBC SW/DL - B' = dead load (hardware format) flight software load to OBC B 'CPE SW - A' = absolute memory flight software load to CPE A' CPE SW - B' = absolute memory flight software load to CPE B' ACIS SW/OBC - A' = OBC-assisted flight software load to ACIS 'ACIS SW - A' = direct (non-OBC-assisted) flight software load to ACIS	ASCII		15

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
	''ACIS SW/OBC - B' = OBC-assisted flight software load to ACIS ACIS SW - B' = direct (non-OBC-assisted) flight software load to ACIS 'AC SW - A' = flight software load to AC –A 'AC SW - B' = flight software load to AC –B 'SIM SW - A' = flight software load to SIM –A 'SIM SW - B' = flight software load to SIM –B 'EIO SW - A' = flight software load to the EPHIN input/output unit A 'IU PROG1 SW - A' = flight software load to IU EEPROM A program 1 'IU PROG2 SW - A' = flight software load to IU EEPROM A program 2 'IU PROG1 SW - B' = flight software load to IU EEPROM program 1 'IU PROG2 SW - B' = flight software load to IU EEPROM program 2 'CTU SW - A' = flight software load to CTU EEPROM A 'CTU SW - B' = flight software load to CTU EEPROM B 'CTU SW - A' = flight software load to CTU EEPROM A 'CTU SW - B' = flight software load to CTU EEPROM B PRT-A – planned real-time command load for OBC-A PRT-B – planned real-time command load for OBC B			
cl_space1	Unused	ASCII		2
cl_num_crit	A count of the number of critical commands in the load (Only specified for ATS or RTS loads, set to zero for other types)	Integer		4
cl_earliest_up	The earliest time the file may be uplinked, in HOSC GMT format	ASCII		17
cl_space2	Unused	ASCII		3
cl_latest_up	The latest time the file may be uplinked, in HOSC GMT format	ASCII		17
cl_space3	Unused	ASCII		3

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cl_num_cmd	A count of the number of 48-bit uplink commands making up the load	Integer		4
cl_severity	The maximum error severity encountered when processing the command data and building the load. Range is 0-4	Integer		4

RECORD LAYOUT:

Record Identifier: Command Load Data Record

Record Format: Fixed

Record Length: 27 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cl_cmd_mnemonic	Unique user friendly mnemonic assigned to the command	ASCII		20
cl_space4	Unused	ASCII		1
cl_cmd_data	Hexadecimal representation of 28 bits of command load data	ASCII		7

3.6 COMMAND LOAD IMAGE

3.6.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (CM)

Data Routinely Accessed By: ONLS, OFLS (CM)

Format Structure: Binary

Delivery Media: Electronic Transfer

3.6.2 CONTENT

The command load image contains a bit map image of a corresponding command load file. It is used by the ONLS for the immediate ‘dump and compare’ of the spacecraft memory affected by a command load.

The command load image contains a header record specifying information about the corresponding load file and the type of the corresponding load as ATS command loads; RTS command loads; OBC, ACIS, CPE, or AC tables; or an OBC, CPE, ACIS, AC, or IU EEPROM flight software load.

The header record is followed by load image data records containing the hexadecimal image of the corresponding command load.

3.6.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Binary

FILE SIZE (ESTIMATED): 150,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Load Image Header Record

Record Format: Fixed

Record Length: 32 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cli_target_proc	Identifier for the target processor for the data contained in the load. Allowed values are: 1 = OBC A; 2 = OBC B; 3 = CPE A 4 = CPE B 5 = IU EEPROM A program 1 6 = Deleted 7 = IU EEPROM B program 1 8 = Deleted 9 =CTU EEPROM A 10 = CTU EEPROM B 11 = AC A RAM 12 = AC A ROM (applies only to dumps) 13 = AC B RAM 14 = AC B ROM (applies only to dumps) 15 = SIM A RAM 16 = SIM A ROM (memory dump only) 17 = SIM B RAM 18 = SIM B ROM (memory dump only) 19 = IEPHIN RAM 20 = EPHIN input/output unit (EIO) A	ASCII		2
cli_type	Identifies the type of image file; the only allowed value is 1 indicating a processor load	ASCII		1

cli_loadgen_time	Identifies the time at which the load was generated (HOSC GMT)	ASCII		21
cli_checksum	Used only in OFP memory loads where: 1 = Compute Checksum when current buffer load completes 0 = Don't checksum after this buffer	ASCII, Numeric, 1 digit		8
cli_VCDU_Counter	The value of the VCDU when the dump started. This field is left blank for loads.	ASCII		8

RECORD LAYOUT:

Record Identifier: Load Image Data Record

Record Format: Variable Length

Record Length (maximum): 2^{32} (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cli_start_loc	Absolute address of the starting location in memory for the load (hexadecimal)	ASCII		8
cli_load_length	Length of the following load data (bytes)	ASCII		8
cli_load_data	Load data in ascending address order (binary)	binary		cli_load_length

3.7 COMMAND SEQUENCE DEFINITIONS

3.7.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: TRW, ASC

Data Routinely Accessed By: OFLS (CM)

Format Structure: Fixed

Delivery Media: Electronic Transfer

3.7.2 CONTENT

The command sequence definition element is used by CM to expand sequence references in the DOT or FOT requests into sequences of spacecraft commands. Each file in the element defines a command sequence. The text in the following section describes the format and syntax for command sequences and how they are used by CM.

3.7.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 150,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Command Sequence Definition Record

Record Format: Fixed

Record Length: 80 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cmd_seq_text	Text line containing command sequence definitions	ASCII		80

3.7.3.1 Command Sequence Statement Definitions

A command sequence request is used to specify the execution of a sequence of commands. The command sequence request consists of a sequence type specification, the command sequence mnemonic, and the associated parameter lists. The statements supported by the command sequence definition syntax are: ATS, RTS, SIMPKT, ACAPKT, EIOPKT, /CMD, RTSLOAD, RTSEXPIRE, AON, AOF, _IF, _ELSE, _ENDIF, _SET, and ACIS. The format of the statements within a command sequence follows the basic formatting rules specified for the observation request data element. Statements are formatted in 80 character text lines. Statements that continue on multiple lines must break at delimiters (commas) between parameters. Statements that do not end with a delimiter at the end of a line are assumed to be complete.

substitution parameter — A parameter on a command sequence or relative time sequence statement referencing a value to be passed to the invoked sequence definition. The value of a substitution parameter can be a symbol, substitution mnemonic, or substitution value of any valid data type.

substitution mnemonic — A mnemonic referencing the value of a substitution parameter or symbol. The mnemonic may reference a substitution parameter or another symbol. Substitution mnemonics must be enclosed in ampersands, &&. Default values can be provided following the substitution mnemonic and must be enclosed in backslashes, \\. Default values are used only if the mnemonic has not be passed by reference on command sequence invocation or defined via a _SET statement. It should be noted that text substitution will be made for every occurrence of a given substitution mnemonic within a given command sequence.

Text substitution of the substitution mnemonic is performed on a physical line before any syntax analysis is performed. This permits the insertion of more than one syntactic element at a time. Note that ampersand characters may only appear on the physical line if it is bracketing a substitution mnemonic. In all cases, after

substitution, the resulting character string must conform to the syntax definition required by the individual statements. However, line continuation past column 80, produced as a result of character substitution, is handled automatically by CM.

If the value passed into a command sequence is a string consisting of 0 to 8 blanks, it is considered a null parameter. When the null parameter is substituted during command sequence expansion, nothing is inserted in place of the &parameter& field. Within the command sequence, the default field may also be blanks. This implies that if the substitution parameter is not supplied, then nothing is substituted for that parameter.

The following are reserved words that cannot be used as substitution mnemonics for any command sequence:

TIME
DELTA
EXPIRE
LOADAFTER
LOADBY
SCS_NUM

Examples: &NUM_ACQ&; &NUMSTAR&; &NUM_ACQ&\5\;

substitution value	The value for a substitution parameter or symbol. The substitution value must be enclosed in ampersands. Examples: &’SSR 1’&
command parameters	Command parameters can be used to provide data to be used during command translation. The data supplied in this manner will be used unchanged in all invocations of the command set definition. Command parameters are referenced directly on the command statement in the form parameter=parameter_value. Parameter values may be Integer, real (decimal), ASCII strings, hex, octal, or binary. String values must be enclosed within single quotes. Hex, octal, and binary data are indicated by enclosing the value in single quotes and preceding the value with X, O, or B, respectively.
symbol	A mnemonic with a defined value created by a _SET statement used for command translation and to control conditional processing. Several symbols as well as command sequence parameters can be concatenated together by using the normal ampersand notation. Symbol substitution is performed prior to statement evaluation so concatenation is available in any statement. After substitution, the resulting string of characters must be less than 24 characters and contain no embedded quotes.

local symbol	A symbol with a defined value created by a _SET statement with the = sign. Local symbols exist only for the duration of a single command sequence. Local symbols are not available to referenced command sequences except by being passed as a substitution parameter. Examples: NUMSTAR=5; NUMSTAR=&NUM_ACQ&; NUMSTAR='5 STARS'
global symbol	A symbol with a defined value created by a _SET statement with the == (double equal sign) notation. Global symbols exist across command sequence boundaries. Global symbols are available to referenced command sequences without being passed as substitution parameters. Examples: SSR==1; SSR==&SSR_PARM&; SSR=='SSR 1'
time	Time can be specified on command, command sequence, and relative time sequence statements. Time parameters are TIME, DELTA, LOADBY, LOADAFTER, and EXPIRE. The TIME parameter can be used only on absolute timed command sequence and command requests from the DOT or from FOT requests. The DELTA parameter is used to specify the relative time spacing of the commands in the command sequence definitions and the relative time sequences. DELTA times can be positive or negative and have the format, plus (+) or minus (-) followed immediately (no blanks) by a relative time (e.g., -03:00:01.00). The LOADBY and LOADAFTER parameters are used to specify uplink restrictions for table, RTS, and software loads. Omitting the LOADBY and LOADAFTER parameters indicates that the load can be scheduled for uplink at any time in the current processing interval. Specifying LOADBY without LOADAFTER defines the latest possible time the uplink must be onboard. Specifying LOADAFTER without LOADBY defines the earliest possible time the uplink may be onboard. Specifying both LOADBY and LOADAFTER completely constrains the uplink window for the load. The EXPIRE parameter indicates when an onboard RTS is no longer required onboard and can be reloaded with another relative time sequence. Time parameter values can take the form of relative, orbit event, and major frame sync time specifications. Relative times are as defined for observation requests.
orbit event time	Orbit event time specification is provided as an option on absolute time specifications. The processing resolves the statement time by referencing the associated time for the specified event in the orbit event file and then applies an optional positive or negative relative time. Orbit event time specifications take the form of TIME=(ORB,orbit_number,orbit_event,+/relative_time).

mfsynch time — Major frame synchronization is provided as an option on absolute time specifications and orbit event time specifications. This processing rounds the absolute time to the next major frame pulse, and then applies an optional positive or negative relative time. Major frame synch time specifications take the form of

TIME=(time_value,MFSYNC,+/-relative_time)
or
TIME=((ORB, orbit_number,orbit_event,+/relative_time)
,MFSYNC,+/-relative_time).

Note that the positive and negative relative times are cumulative for major frame synchronization to orbit event times. The orbit event time is resolved including any relative time, the time is rounded to the next major frame pulse, and the major frame synch relative time is applied.

comment — Comments are freeform text supplied by the developer of the command sequence definition. Comments are indicated by an exclamation point, !. All text on any line following an exclamation point is interpreted as a comment. Comments can appear anywhere in the command sequence. Comments can be used to document the function of each line in the command sequence. Alternately, a line can contain only comments.

3.7.3.2 ATS Statement Syntax

ATS, mnemonic,DELTA=relative time[,substitution parameter=substitution_mnemonic|substitution_value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command sequence definition name	Mnemonic	n/a	1-20 char (TBR)	n/a
	DELTA	Invocation time	Relative time	GMT	n/a	n/a
	substitution parameter	Substitution Parameter for Command Sequence Invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution mnemonic, or Symbol	n/a	n/a	n/a

ATS

The ATS statement is used to reference a predefined sequence of commands that are stored in the absolute time sequence onboard (daily load) and executed at an absolute time

based on the spacecraft clock. Processing of the current command sequence definition is halted and processing begins with the first statement in the referenced command sequence definition. Processing of the included command sequence proceeds normally until the last statement which then returns processing back to the statement following the ATS statement. A relative timetag on the next statement following the ATS statement is timed from the ATS statement.

mnemonic — The mnemonic specifies the name of the command sequence definition file containing a sequence of commands to be included in the onboard absolute time sequence command load. This parameter must reference a command sequence definition defined in the command sequence definition data element of the ODE. This is a required parameter.

DELTA — The DELTA parameter specifies the relative time after the last preceding ATS, RTS, CMD, or ACIS statement at which the command sequence definition will be invoked. The preceding statement is determined only from the statements from within any one command sequence definition file. If the first command or sequence definition within the referenced command sequence has no timetag, it will be tagged with the time resolved from the DELTA parameter. If the first command or sequence definition within the referenced command sequence definition has a DELTA time timetag, it will be tagged with the time resolved by adding the DELTA time to the time resolved from the DELTA parameter value. If DELTA is not specified, then the command sequence definition will be invoked at the same time as the previous ATS, RTS, CMD, or ACIS statement. This is an optional parameter.

Substitution parameter — The substitution parameters specify values to be passed to the referenced command sequence definition. Any value used within the referenced command sequence definition must be supplied via a substitution parameter or have a default value defined within the command sequence definition.

3.7.3.3 RTS Statement Syntax

RTS, mnemonic, SCS_NUM=integer,DELTA=relative time

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Relative time sequence name	Mnemonic	n/a	1-20 char (TBR)	n/a
*	SCS_NUM	Number of SCS to be loaded	Integer	n/a	1-256	n/a
	DELTA	Invocation time	Relative time	GMT	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	substitution parameter	Substitution Parameter for Command Sequence Invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution parameter, or Symbol	n/a	n/a	n/a

RTS

The RTS statement is used to reference a predefined sequence of commands that are stored in relative time sequence onboard and executed when referenced from the onboard absolute time sequence (daily load). The relative time sequence must be already loaded onboard at the time the RTS statement is invoked. Processing of the current command sequence definition is halted and processing begins with the first statement in the referenced command sequence definition. The command and sequence request timetags within the relative time sequence definition are resolved based on the RTS statement time to perform command timing constraint checking. A relative timetag on the next statement following the RTS statement is timed from the RTS statement.

mnemonic — The mnemonic specifies the name of the relative time sequence definition file containing a sequence of commands that has been loaded onboard. This parameter must reference a relative time sequence definition defined in the relative time sequence definition data element of the ODE. This is a required parameter.

SCS_NUM — The SCS_NUM parameter specifies the number of the SCS in OBC memory to be invoked by the RTS statement.

DELTA — The DELTA parameter specifies the relative time after the last preceding ATS, RTS, CMD, or ACIS statement at which the relative time sequence definition will be invoked. The preceding statement is determined only from the statements from within any one command sequence file. If the first command or sequence definition within the referenced relative time sequence has no timetag, it will be tagged with the time resolved from the DELTA parameter. If the first command or sequence definition within the referenced relative time sequence definition has a DELTA time timetag, it will be tagged with the time resolved by adding the DELTA time to the time resolved from the DELTA parameter value. If DELTA is not specified, then the command sequence definition will be invoked at the same time as the previous ATS, RTS, CMD, or ACIS statement. This is an optional parameter.

3.7.3.3A SIMPKT Statement Syntax

SIMPKT, mnemonic,DELTA=relative_time [,substitution_parameter = substitution_mnemonic|substitution_value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command sequence definition name	Mnemonic	n/a	1-20 char	n/a
	DELTA	Invocation time	Relative time	n/a	n/a	n/a
	substitution parameter	Substitution parameter for command sequence invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution mnemonic, or Symbol			

SIMPKT

The SIMPKT statement is used to reference a predefined sequence of commands that are stored as a packet in the absolute time sequence onboard (daily load) and executed at an absolute time based on the spacecraft clock. Processing of the current command sequence definition is halted and processing begins with the first statement in the referenced command sequence definition. Each statement of the referenced command sequence proceeds normally (as for ATS statement processing) with the following exceptions:

a SIM packet header is added to the command stream prior to the commands within the referenced command sequence definition

the packet command count is the count of the commands within the referenced command sequence definition. This may not exceed the maximum command count for SIM packets

DELTA time parameters on the statements within the referenced command sequence definition are used for command timing constraint checks only, no absolute time wait or relative time wait commands are added to the command stream as the result of DELTA time parameters

When the last statement is processed, processing returns back to the statement following the SIMPKT statement. A relative timetag on the next statement following the SIMPKT statement is timed from the SIMPKT statement.

3.7.3.3B ACAPKT Statement Syntax

ACAPKT, mnemonic,DELTA=relative_time [,substitution_parameter = substitution_mnemonic|substitution_value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command sequence	Mnemonic	n/a	1-20 char	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
		definition name				
	DELTA	Invocation time	Relative time	n/a	n/a	n/a
	substitution parameter	Substitution parameter for command sequence invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution mnemonic, or Symbol			

ACAPKT

The ACAPKT statement is used to reference a predefined sequence of commands that are stored as a packet in the absolute time sequence onboard (daily load) and executed at an absolute time based on the spacecraft clock. Processing of the current command sequence definition is halted and processing begins with the first statement in the referenced command sequence definition. Each statement of the referenced command sequence proceeds normally (as for ATS statement processing) with the following exceptions:

a ACA packet header is added to the command stream prior to the commands within the referenced command sequence definition

the packet command count is the count of the commands within the referenced command sequence definition. This may not exceed the maximum command count for ACA packets

DELTA time parameters on the statements within the referenced command sequence definition are used for command timing constraint checks only, no absolute time wait or relative time wait commands are added to the command stream as the result of DELTA time parameters

When the last statement is processed, processing returns back to the statement following the ACAPKT statement. A relative timetag on the next statement following the ACAPKT statement is timed from the ACAPKT statement.

3.7.3.3C EIOPKT Statement Syntax

EIOPKT, mnemonic,DELTA=relative_time [,substitution_parameter = substitution_mnemonic|substitution_value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command sequence definition name	Mnemonic	n/a	1-20 char	n/a
	DELTA	Invocation time	Relative time	n/a	n/a	n/a
	substitution parameter	Substitution parameter for command sequence invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution mnemonic, or Symbol			

EIOPKT

The EIOPKT statement is used to reference a predefined sequence of commands that are stored as a packet in the absolute time sequence onboard (daily load) and executed at an absolute time based on the spacecraft clock. Processing of the current command sequence definition is halted and processing begins with the first statement in the referenced command sequence definition. Each statement of the referenced command sequence proceeds normally (as for ATS statement processing) with the following exceptions:

a EIO packet header is added to the command stream prior to the commands within the referenced command sequence definition

the packet command count is the count of the commands within the referenced command sequence definition. This may not exceed the maximum command count for EIO packets

DELTA time parameters on the statements within the referenced command sequence definition are used for command timing constraint checks only, no absolute time wait or relative time wait commands are added to the command stream as the result of DELTA time parameters

When the last statement is processed, processing returns back to the statement following the EIOPKT statement. A relative timetag on the next statement following the EIOPKT statement is timed from the EIOPKT statement.

3.7.3.4 CMD Statement

/CMD, mnemonic,DELTA=relative time[,substitution parameter=substitution_mnemonic|substitution_value]

/ mnemonic,DELTA=relative time[,substitution parameter= substitution_mnemonic|substitution_value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command mnemonic	Mnemonic	n/a	1-20 char (TBR)	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	DELTA	Invocation time	Relative time	GMT	n/a	n/a
	substitution parameter	Substitution Parameter for Command Translation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution parameter, or Symbol	n/a	n/a	n/a

CMD

A command statement is used to specify the execution of a single command defined in the command database. Each command statement references one and only one command mnemonic. The general form of a command statement consists of an optional label, a slash (/), an optional keyword (CMD), the command mnemonic, the associated optional data consisting of the substitution parameter list, and time specification. Every command will have an associated command mnemonic that is defined in the command database. For discrete commands, the command performs a single function and there is no additional substitution parameter data associated with the command. For serial data commands, the command is followed by the substitution parameter data which apply to the command. The substitution parameters are associated with the command parameter mnemonics in the command database.

mnemonic — The mnemonic parameter specifies the mnemonic of the command to be retrieved from the command database. This parameter must reference a command defined in the command database of the ODB. This is a required parameter.

DELTA — The DELTA parameter specifies the relative time after the last preceding ATS, RTS, CMD, or ACIS statement at which the command will be invoked. The preceding statement is determined only from the statements from within any one command sequence definition file. If the DELTA parameter is not specified, then the command will be time-tagged at the same time as the previous ATS, RTS, CMD, or ACIS statement. This is an optional parameter.

Substitution parameter — The substitution parameters specify values to be used in building the referenced command. Any modifiable parameter used by the referenced command definition must be supplied via a substitution parameter or have a default value defined within the command sequence definition.

3.7.3.5 RTSLOAD Statement

RTSLOAD, mnemonic,SCS_NUM=integer[,LOADBY=relative time] [,LOADAFTER= relative time] [,EXPIRE=relative time] [,substitution parameter= substitution_mnemonic|substitution _value]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Relative time sequence name	Mnemonic	n/a	1-20 char (TBR)	n/a
*	SCS_NUM	Number of SCS to be loaded	Integer	n/a	1-256	n/a
	LOADBY	Time by which relative time sequence must be loaded onboard	Relative time	GMT	n/a	n/a
	LOADAFTER	Time after which relative time sequence may be loaded onboard	Relative time	GMT	n/a	n/a
	EXPIRE	Time after which relative time sequence can be used as the destination of another RTSLOAD statement	Relative time			
	substitution parameter	Substitution Parameter for Command Sequence Invocation	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution parameter, or Symbol	n/a	n/a	n/a

RTSLOAD

The RTSLOAD statement is used to reference an RTS definition to be built and loaded into an OBC stored command sequence. The relative time sequence is obtained from the ODE, translated into binary format, and formatted for uplink to the OBC. The RTS must be defined in the Relative Time Sequence data element of the ODE as defined by the mnemonic parameter. The parameters on the RTSLOAD statement must match exactly the names of the parameters in the relative time sequence definition file.

mnemonic — The mnemonic parameter specifies the name of the relative time sequence definition file containing the set of commands to be loaded onboard. This parameter must reference a relative time sequence definition defined in the relative time sequence definition data element of the ODE. This is a required parameter.

SCS_NUM — The SCS_NUM parameter specifies the number of the destination SCS where the relative time sequence is to be loaded in OBC memory. This is a required parameter.

LOADBY — The LOADBY parameter specifies the time by which the relative time sequence must be loaded in the final destination. If this parameter is omitted, the relative time sequence must be loaded no later than the end of the command load generation run end time. This is an optional parameter.

LOADAFTER — The LOADAFTER parameter specifies the time after which the relative time sequence can be scheduled for uplink to the table destination. If this parameter is omitted, the relative time sequence may be loaded any time after the command load generation run start time. This is an optional parameter.

EXPIRE — The EXPIRE parameter specifies the time of the last use of the relative time sequence onboard. After this time, the stored command sequence identified by the SCS_NUM parameter can be used as the destination of an RTSLOAD statement. If this parameter is omitted, the relative time sequence cannot be reused until an RTSEXPIRE statement is issued. This is an optional parameter.

Substitution parameter — The substitution parameters specify values to be passed to the referenced relative time sequence definition. Any value used within the referenced relative time sequence definition must be supplied via a substitution parameter or have a default value defined within the relative time sequence definition or command definition.

3.7.3.6 RTSEXPIRE Statement

RTSEXPIRE, mnemonic,SCS_NUM=integer,EXPIRE=relative time

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Relative time sequence name	Mnemonic	n/a	1-20 char (TBR)	n/a
*	SCS_NUM	Number of SCS to be loaded	Integer	n/a	0-256	n/a
*	EXPIRE	Time after which relative time sequence can be used as the destination of another RTSLOAD statement	Relative time			

RTSEXPIRE

The RTSEXPIRE statement is used to indicate that a relative time sequence is no longer required onboard and can be loaded with another RTS. The RTSLOAD statement cannot reference an onboard SCS via the SCS number that has not been expired by either an EXPIRE time on the original RTSLOAD statement, or an RTSEXPIRE statement.

mnemonic — The mnemonic parameter specifies the name of the relative time sequence definition file containing the set of commands that have

	been loaded onboard. This parameter must reference a relative time sequence definition defined in the relative time sequence definition data element of the ODE. This is a required parameter.
SCS_NUM	The SCS_NUM parameter specifies the number of the destination SCS where the relative time sequence is to be loaded in OBC memory. This is a required parameter.
EXPIRE	The EXPIRE parameter specifies the time of the last use of the relative time sequence onboard. After this time, the stored command sequence identified by the SCS_NUM parameter can be used as the destination of an RTSLOAD statement. This is a required parameter.

3.7.3.7 AON Statement

AON

The AON statement defines the start of an atom group of commands. Atom groups identify critical sequences of commands which must execute without potential of a load uplink failure affecting partial sequence execution. The commands within an atom group will not cross load boundaries unless the load break is due to a user-specified forced load break. Defined especially for use within sequence definitions, there are no particular restrictions on the definition of atom groups. Sequence definitions may be nested within atom groups, and the entire nested sequence will be treated as a single atom group. The AON statement allows no parameters. The AON statement must be followed by the AOFF statement within the command sequence definition. Atom groups cannot cross command sequence boundaries.

3.7.3.8 AOFF Statement

AOFF

The AOFF statement defines the end of an atom group of commands. The AOFF statement allows no parameters. The AOFF statement must be preceded by the AON statement within the command sequence definition.

3.7.3.9 IF Statement

_IF value=<>value

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	value	If Test Comparison Value	Real, Integer, Hex, Octal, ASCII string, Mnemonic, Substitution parameter, or Symbol	n/a	n/a	n/a

_IF

The _IF statement is the first statement in a compound conditional construct requiring more than one statement to construct a meaningful request. The sequence of commands

executed may be conditionally altered by the _IF statement. The expression specified following the _IF is evaluated to true or false, and, if true, causes the conditional translation of command sequence statements up to the next _ELSE or _ENDIF statement. If the expression is false, evaluation of the command sequence proceeds following the _ELSE statement or following the _ENDIF if no _ELSE clause is provided. The _IF statements may be nested to any level. Expression evaluation allows for ASCII strings supplied by symbols (local or global) or command sequence parameter mnemonics to be lexically compared equal, or not equal to an integer, real number, hex, octal, or quoted string of ASCII characters. This string of ASCII characters may not contain embedded quote marks or ampersands which are reserved for substitution purposes. The _IF keyword cannot be inserted by substitution because it is needed prior to substitution evaluation. The _IF statement must be followed by an _ELSE and an _ENDIF statement or an _ENDIF statement. Compound conditional constructs cannot cross command sequence definition boundaries.

An example of the complete syntax for a compound conditional construct using all three statements is:

_IF X=1

/CMD,MNEMONIC1,DELTA=00:04:00

_ELSE

/CMD,MNEMONIC2,DELTA=00:00:02

_ENDIF

An example of the complete syntax for a compound conditional construct using only two of the construct statements is:

_IF X=1

/CMD,mnemonic1,TIME=absolute time

_ENDIF

3.7.3.10 ELSE Statement

_ELSE

The _ELSE statement is an optional statement in a compound conditional construct requiring more than one statement to construct a meaningful request. The _ELSE statement specifies the set of statements to be evaluated if the expression following the _IF statement is evaluated to be false. The _ELSE statement allows no parameters. The _ELSE keyword cannot be inserted by substitution because it is needed prior to substitution evaluation. The _ELSE statement must be preceded by an _IF statement and followed by an _ENDIF statement. Compound conditional constructs cannot cross command sequence definition boundaries.

3.7.3.11 ENDIF Statement

_ENDIF

The _ENDIF statement is the final statement in a compound conditional construct requiring more than one statement to construct a meaningful request. The _ENDIF statement specifies the end of the set of command sequence statements to be evaluated for the true case if the compound conditional construct contains no _ELSE statement or for the false case if the compound conditional construct contains an _ELSE statement. The _ENDIF statement allows no parameters. The _ENDIF keyword cannot be inserted by substitution because it is needed prior to substitution evaluation. The _ENDIF statement must be preceded by an _IF statement or an _ELSE statement. Compound conditional constructs cannot cross command sequence definition boundaries.

3.7.3.12 SET Statement

_SET SYMBOL=|==value

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	SYMBOL	Global or local symbol name	ASCII string	n/a	n/a	n/a

_SET

The _ SET statement is used to initialize the value of a local or global symbol to a specified value. The _SET statement allows the creation of symbols within command sequences which are usable for parameter substitution or _IF statement expression evaluation just like command sequence parameter mnemonics. Symbols local to a command sequence, or global to all command sequences, may be created. Global symbols are created by using a _SET statement with the == (double equal sign) notation. Local symbols are created with a single = sign.

3.7.3.13 DELETE Statement

_DELETE SYMBOL

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	SYMBOL	Global symbol name	ASCII string	n/a	n/a	n/a

_DELETE

The _DELETE statement specifies a symbol to be removed from the global symbol table. A command sequence can be used which only contains _SET and/or _DELETE statements.

3.7.3.14 ACIS Statement

ACIS, mnemonic, DELTA=relative time

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	mnemonic	Command mnemonic	mnemonic	n/a	1-20 char	n/a
	DELTA	Invocation time	relative time	GMT	n/a	n/a

ACIS

A table statement used to specify an ACIS parameter block defined in the ACIS table data element (section 3.43). Each ACIS statement references one and only one ACIS table definition. The referenced ACIS table definition will be included in the load at the time resolved from the DELTA time parameter (if any) or at the time of the previous statement if no DELTA parameter is specified.

mnemonic The mnemonic specified the name of the ACIS table definition containing the parameter block to be uplinked. This parameter must reference a table identification tag in the ACIS table data element in the ODE.

DELTA The DELTA parameter specifies the relative time after the last preceding ATS, RTS, CMD, or ACIS statement at which the ACIS parameter block will be invoked. Then preceding statement is determined only from the statements from within any one command sequence file. If DELTA is not specified, then the ACIS parameter block will be invoked at the same time as the previous ATS, RTS, CMD, or ACIS statement. This is an optional parameter.

3.7.3.15 Sample Command Sequence Definition

A command sequence SETOBS may be defined as:

```
!  Command sequence definition for controlling wheel activation
!  Developed by S. Kwong,  12/21/96
_IF &OPTION& = 1
!  using default wheel configuration 1
/CMD,MNEMONIC1        ! setup command
/CMD,SETWHEEL,S=&SET&/1/,FILTER=&FIL&/3/,DELTA=00:00:02  !  wheel command
_ELSE
!  using default wheel configuration 2
/CMD,MNEMONIC2       ! setup command
/CMD,SETWHEEL,S=&SET&/2/,FILTER=&FIL&/3/,DELTA=00:00:02  !  wheel command
_ENDIF
```

The command SETWHEEL includes two command parameter mnemonics, S and FILTER whose sequence definition values are substitution mnemonics SET and FIL, with defaults of 1 and 3, respectively for the IF true block and defaults 2 and 3, respectively for the IF false block. The SETWHEEL, S and FILTER mnemonics are all defined in the command database.

If, the SETOBS command sequence is invoked as

```
ATS, SETOBS,OPTION=1,TIME=(ORB,2271,XSAA1)
```

then the command SETWHEEL, as indicated below will contain the default value of 1 for parameter S and the default value of 3 for parameter FILTER

```
/CMD,SETWHEEL,S=1,FILTER=3
```

If, the SETOBS command sequence is invoked as

ATS, SETOBS,OPTION=2,TIME=(ORB,2271,XSAA1)

then the command SETWHEEL, as indicated below will contain the default value of 2 for parameter S and the default value of 3 for parameter FILTER

/CMD,SETWHEEL,S=2,FILTER=3

If, the SETOBS command sequence is invoked as

ATS, SETOBS,OPTION=1,SET=4,TIME=(ORB,2271,XSAA1)

then the command SETWHEEL, as indicated below will contain the substitution value of 4 for parameter S and the default value of 3 for parameter FILTER

/CMD,SETWHEEL,S=4,FILTER=3

3.8 CONFIGURATION REFERENCE

3.8.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	TRW
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	ASCII, Sequential
Delivery Media:	Electronic Transfer

3.8.2 CONTENT

The configuration reference data element contains a set of spacecraft telemetry MSIDs for which telemetry verifier values from the operational command database (OCDB) are to be placed in the configuration snapshot. A configuration reference record may contain a command mnemonic, the corresponding telemetry verifier mnemonic, and the expected value of the telemetry MSID for that command mnemonic. If no telemetry verifier is defined in the OCDB for a command mnemonic, then an expected value must be provided in the configuration reference in order to have the OFLS report a predicted telemetry value for that command mnemonic. If a predicted value is provided and the command mnemonic left blank, the predicted value will be used as a default value for the telemetry MSID value.

Multiple configuration references may exist for different mission phases such as orbital verification and for different spacecraft events such as eclipse season. Each configuration reference is a file with one record for each telemetry mnemonic.

3.8.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 800 (Bytes)

RECORD LAYOUT:

Record Identifier: Configuration Reference Record

Record Format: Fixed

Record Length: 65 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
ref_telem_msid	Telemetry MSID to be included in the Configuration Reference Report	ASCII		20
ref_cmd_mnem	Command mnemonic to use this MSID as a verifier for (can be blank)	ASCII		20
ref_msid_value	MSID value to use for verification of the listed command mnemonic, or as the default MSID value if the command mnemonic is blank (can be blank)	ASCII		25

3.9 CONFIGURATION SNAPSHOT

3.9.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (CM)

Data Routinely Accessed By: ONLS (Mission Comp), OFLS (CM)

Format Structure: ASCII, Sequential

Delivery Media: Electronic Transfer

3.9.2 CONTENT

The configuration snapshot contains a tabular listing of telemetry MSIDs and their expected state or range (low, high) at the start of each DSN contact, built automatically based on the DOT, FOT requests, and the configuration reference.

Each configuration snapshot is a file with the first record specifying the start time of the scheduled DSN contact followed by multiple records, one for each telemetry MSID and expected value.

3.9.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 70,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Configuration Snapshot Contact Time Record

Record Format: Fixed

Record Length: 17 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cnfg_contact_start	Time of predicted start of DSN contact in HOSC GMT format	ASCII		17

RECORD LAYOUT:

Record Identifier: Configuration Snapshot MSID Record

Record Format: Fixed

Record Length: 45 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cnfg_msid	Telemetry MSID (mnemonic)	ASCII		20
cnfg_msid_value	Expected value for MSID	ASCII		25

3.10 CONSTRAINTS

3.10.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: TRW, ASC, CSC

Data Routinely Accessed By: OFLS

Format Structure: ASCII, Sequential

Delivery Media: Electronic Transfer

3.10.2 CONTENT

The AXAF constraints file contains parameters used by the OFLS to define spacecraft, subsystem, and basic mission constraints including: maneuver, aspect camera, pointing, SI, power, LGA, SSR, and communications constraints. Although this element will be maintained by the FOT, initial values for the data parameters contained in this element will be provided jointly by TRW (Spacecraft developer), ASC (Science Instrument Team), and CSC (OFLS developer).

3.10.3 FORMAT DESCRIPTION

The constraint parameters are grouped into related records as shown in the following section. Each record is formatted as an ANSI standard FORTRAN namelist as defined for Characteristics data element in section 3.2.

3.10.3.1 Data Records

This section describes the Constraints data parameters requested by the OFLS (element format provider). The information described below were reviewed by TRW (element data provider). Due to discrepancies between algorithms adapted by OFLS and those used by Spacecraft software, initial values will be provided by TRW and CSC as specified in the initial value provider column in the following table. A cross-reference of the parameter to the OFLS Software Design Specification (AMO-2310) document section is also provided for additional reference.

Each record is formatted as an ANSI standard FORTRAN namelist.

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 500,000 (Bytes)

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
1	odb_sun_maneuver_constraint	Defines sun pointing constraint during spacecraft maneuvers (degrees).	Real		8	TRW PCAD	4.1.3.8 (Check Maneuver Path Sun Constraints)	γ_0
2	odb_sa1slew	Defines minimum slew angle through which SA-1 can slew (minimum angular motion) (degrees).	Real		8	TRW PCAD		
3	odb_sacon	Defines SA timing constants (1) = minimum SA slew duration term (seconds) (2) = slope term (degrees per second) (3) = number of degrees covered in one small pulse (degrees)	Real	3	24	TRW PCAD		

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
4	odb_sa2slew	Defines minimum slew angle through which SA-2 can slew (minimum angular motion) (degrees).	Real		8	TRW PCAD		
5	odb_saoff	Defines SA offset from perpendicularity to sunline (degrees) (index SA ID)	Real	2	16	TRW PCAD		
6	odb_condur	Defines SA slew angle above which a slew duration is calculated for use rather than a constant value (degrees).	Real		8	TRW PCAD		
7	odb_maxacc	Defines maximum angular acceleration (degrees per second^2; index: 1 = reaction wheel case (6, 4 reaction wheels), 2 = slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (calculate required maneuver)	α_m
8	odb_mintime	Defines minimum time between maneuvers on reaction wheels (seconds).	Relative time		8	TRW PCAD		
9	odb_maxvel	Defines maximum angular velocity (degrees per second; index: 1 = reaction wheel case (6, 4 reaction wheels), 2 = slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (calculate required maneuver)	v_m

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
10	odb_deltamin	Defines minimum jerk pulse width (seconds; index: 1 = reaction wheel case (6, 4 reaction wheels), 2 = slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (calculate required maneuver)	N/A
11	odb_sc_settle	Defines settling time for spacecraft slews (seconds; index: slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (Calculate Required Maneuver)	N/A
12	odb_sa_settle	Defines settling time for SA slews (seconds; index: SA ID)	Relative time	2	16	TRW PCAD	TBD	TBD
13	Deleted							
14	odb_ epslnmax	Defines maximum constant acceleration interval (seconds; index: 1 = reaction wheel case (6, 4 reaction wheels), 2 = slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (calculate required maneuver)	e_m
15	odb_jerkmax	Defines maximum jerk (degrees per second3; index: 1 = reaction wheel case (6, 4 reaction wheels), 2 = slew type (RCS, on-orbit RWA, transfer orbit RWA))	Real	2x3	48	TRW PCAD	4.1.3.7 (calculate required maneuver)	J_m
16	odb_numrw	Defines number of reaction wheels in use.	Integer		4	TRW PCAD	4.1.3.7 (calculate required maneuver)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
17	odb_min_maneuver_angle	Defines the minimum angular change allowed during a maneuver (degrees)	real		8	TRW PCAD		
18	odb_min_ndg_stars	Defines the minimum number of stars required to retain tracking lock during a nudge mode attitude change	Integer		4	TRW PCAD	N/A	
19	odb_max_maneuver_angle	Defines the maximum angular change for a maneuver without requiring roll constraint checking and possible maneuver segmentation (degrees)	Real		8	TRW PCAD	4.1.3.10 (Calculate Maneuver Path Roll Constraint)	N/A
20	odb_man_roll_limit	Defines the maximum off-nominal roll value allowed at the start and end of a maneuver for maneuvers that are short enough (odb_max_maneuver_angle) to skip roll constraint checking (degrees)	Real		8	TRW PCAD	4.1.3.10 (Calculate Maneuver Path Roll Constraint)	N/A
21	odb_num_suntox_ang	Defines the number of angles in the Sun to x-axis angle table, odb_suntox_ang, used during maneuver constraint checking. (index: 1 = number of angles for positive roll (in the spacecraft frame) 2 = number of angles for negative roll)	Integer	2	8	TRW SE	4.1.3.10 (Calculate Maneuver Path Roll Constraint)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
22	odb_suntox_ang	Sun to x-axis angle table during maneuvers. Threshold limits defining ranges of Sun to x-axis angles (pitch) that determine which value is selected from the maximum deviation from nominal roll during a maneuver table, odb_max_roll_dev. Values must be in increasing order. (degrees) Index: 1 = Sun to x-axis angles, 2 = roll (positive roll (in spacecraft frame), negative roll))	Real	30x2	480	TRW SE	4.1.3.10 (Calculate Maneuver Path Roll Constraint)	N/A
23	odb_max_roll_dev	Maximum deviation from nominal roll during a maneuver at the Sun to x-axis angles tabulated in odb_suntox_ang. (degrees; index 1 = maximum roll deviation, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TRW SE	4.1.3.10 (Calculate Maneuver Path Roll Constraint)	N/A
23A	odb_minsep	Defines the minimum separation between the end of one maneuver and the start of the next maneuver (seconds).	Integer		4	TRW PCAD		

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
	ODE_Roll_Constraints		**Fixed**		**80**			
24	odb_n_sunx_ang	Number of angles in Sun to x-axis angle table, odb_sunx_ang (Index: 1 = number of angles for positive roll (in the spacecraft frame) 2 = number of angles for negative roll)	Integer	2	8	TBD		To Be Provided
25	odb_sunx_ang	Sun to x-axis angle table. Threshold limits defining ranges of Sun to x-axis angles (pitch) that determine which value is selected from the maximum deviation from nominal (zero) roll table, odb_rol_dev_max. Values must be increasing order. (degrees; index: 1 = Sun to x-axis angle, 2 = roll (positive roll (in the spacecraft frame), negative roll)	Real	30x2	480	TBD		To Be Provided

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
26	odb_rol_dev_max	Maximum deviation from nominal (zero) roll at the Sun to x-axis angles tabulated in odb_sunx_ang (degrees; index: 1 = maximum roll deviation, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD		To Be Provided
27	Deleted							
28	odb_rol_dur_max	Maximum deviation from nominal roll for attitudes with a off-nominal roll timeline defined in odb_tim_durmax, odb_time_rec_min, odb_per_recdur, and odb_tim_adddur. Values are for the Sun to X-axis angles tabulated in odb_sunx_ang. (degrees; index: 1 = Sun to X-axis angle, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD	4.1.3.4 Check Deviation from Nominal Roll	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
29	odb_tim_durmax	Maximum duration of off-nominal roll excursion at the Sun to SA normal incidence angles tabulated in odb_sunsa_ang (seconds; index: 1 = maximum off-nominal roll deviation, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD		To Be Provided
30	odb_tim_recmin	Minimum recovery time after off-nominal excursion at the Sun to SA normal incidence angles tabulated in odb_sunsa_ang (seconds; index: 1 = minimum recover time, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD		To Be Provided
31	odb_per_recdur	Percentage of off-nominal roll duration required for recovery at the Sun to SA normal incidence angles tabulated in odb_sunsa_ang (index: 1 = percentage recovery time, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD		To Be Provided

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
32	odb_tim_adddur	Additional duration of required recovery period for off-nominal roll excursion at the Sun to SA normal incidence angles tabulated in odb_sunsa_ang (seconds; index: 1 = duration of additional recovery time, 2 = roll (positive roll (in the spacecraft frame), negative roll))	Real	30x2	480	TBD		To Be Provided
	ODE_Attitude_Hold_Constraints		**Fixed**		**80**			
33	odb_ang_earth	Defines minimum allowed angle between the spacecraft boresight and the Earth limb (degrees).	Real		8	TRW SE	4.1.11 (Check DOT constraints)	ε
34	odb_ang_sun	Defines the allowed angle between the spacecraft boresight and the sun (degrees; index: 1 = minimun angle with sunshade door open, 2 = minimum angle with sunshade door closed, 3 = maximum angle)	Real	2	16	TRW SE	4.1.4.3.1 (check minimum sun pointing constraint); 4.1.4.3.2 (check maximum sun pointing constraint)	γ_0
35	odb_anglun	Defines minimum allowed angle between the spacecraft boresight and the Moon limb (degrees).	Real		8	TRW SE	4.1.4.2 (check moon occultation)	N/A
36	odb_angss	Defines the minimum allowed angle between the spacecraft boresight and the planets (degrees; 1=Mercury, 2=Venus, 3=Mars, 4=Jupiter, 5=Saturn, 6=Uranus, 7=Neptune, 8=Pluto).	Real	8	64	TRW SE	4.1.4.4 (check solar system object occultation)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
37	odb_max_dither_amp	Defines the maximum allowed dither amplitude (degree; index: 1=pitch, 2= yaw s) *WARNING*: **OFLS 9.x does not use this value for constraint checks. Checking is performed in MTRAN using values in syntax rules.**	Real	2	16	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
38	odb_max_dither_freq	Defines the maximum allowed dither frequency (degrees/second; index: 1=pitch, 2= yaw)) *WARNING*: **OFLS 9.x does not use this value for constraint checks. Checking is performed in MTRAN using values in syntax rules.**	Real	2	16	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]
39	odb_max_dither_ph	Defines the maximum allowed dither phase (degrees; index: 1=pitch, 2= yaw) *WARNING*: **OFLS 9.x does not use this value for constraint checks. Checking is performed in MTRAN using values in syntax rules.**	Real	2	16	TRW PCAD	4.1.3.12.n (Select Acquisition Stars, Guide Stars and Fiducial Lights)	N/A[1]

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
39A	odb_max_dither_rate	Defines the maximum allowed dither rate computed as a function of amplitude and frequency (degrees/second) **Note:** The dither rate is calculated using amplitude and frequency supplied in the OR or ER and compared to this constraint value during ORE/ER translation in MTRAN.	Real		8	TRW PCAD	N/A	
40	odb_ang_broll	Defines the maximum angle between the sun and the negative x-axis at which the spacecraft roll angle is restricted (degrees)	Real		8	TRW SE	4.1.3.4 (Calculate Maneuver Path Roll Constraint)	N/A
	ODE_Object_Ephemeris_Constraints		**Variable**		**80**			
41	odb_num_kepl_obj	Number of celestial objects specified by Keplerian orbital elements	Integer		4	TBD	4.1.4.5 (check celestial avoidance regions)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
42	odb_obj_orbit	Six Keplerian orbital elements for the celestial object (index: 1=semi-major axis (kilometers), 2=eccentricity, 3=inclination (degrees), 4=right ascension of the ascending node (degrees), 5=argument of perigee (degrees), 6=mean anomaly (degrees)). If the object is specified by Keplerian orbital elements, odb_obj_pos cannot be specified.	Real	6x20	960	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
43	odb_kepl_obj_id	Identifier for objects specified by Keplerian orbital elements	ASCII	20	160	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
44	odb_kepl_obj_ep	Epoch time for Keplerian orbital elements	ASCII	20	420	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
45	odb_num_pos_obj	Number of celestial objects specified by celestial coordinates	Integer		4	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
46	odb_obj_pos	Object position in celestial coordinates: right ascension and declination (degrees). If the object is specified by celestial coordinates, odb_obj_orbit cannot be specified.	Real	2x20	320	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
47	odb_pos_obj_id	Identifier for object.	ASCII	20	160	TBD	4.1.4.5 (check celestial avoidance regions)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
48	odb_xray_angle	Defines minimum allowed angle between spacecraft boresight and bright x-ray objects (degrees; index: 1 = x-ray object).	Real	20	160	TBD	4.1.4.5 (check celestial avoidance regions)	N/A
	ODE_AC_Constraints		**Fixed**		**80**			
49	odb_rateac	Defines AC maximum tracking rate (arc-sec per second).	Real		8	TRW PCAD	TBD	TBD
50	odb_iac	Defines AC usability indicator (=.TRUE., Hardware usable).	Logical		4	TRW PCAD	4.3.2.3.6.1 (simulate ACA)	N/A
	ODE_PCAD_Constraints		**Fixed**		**80**			
51	odb_css	Operational CSS indicator for each CSS, =.TRUE. If CSS is operational	Logical	4	8	TRW PCAD	4.3.1.1.4 (Process Coarse Sun Sensor Data), 4.3.2.3.3	N/A
52	odb_iru_chan_select	Identifies currently active IRU channel combination (1-126)	Integer		4	TRW PCAD	4.3.1.1.1, 4.3.2.3.1 (Sigma Edit Range Check and Gap Fill)	N/A
53	odb_fss_head_id	Identifies currently active FSS (1 or 2)	Integer		4	TRW PCAD	4.3.2.3.4 (simulate FSS)	N/A
54	odb_esa_head_id	Identifies currently active ESA (1 or 2)	Integer		4	TRW PCAD	4.3.2.3.5 (Simulate ESA)	N/A
55	odb_max_mom	Defines the maximum allowed spacecraft momentum in the spacecraft body frame (kilogram - meters^2 /second; index: 1= minimum, 2=maximum)	Real	3x2	48	TRW PCAD	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
55A	odb_esa	Defines which ESAs are operational, =.TRUE.,if ESA is operational; index = ESA	Logical	2	8	TRW PCAD	4.1.3.5 (Position Target on Optical Axis)	N/A
55B	odb_mom_dump	Defines the desired spacecraft momentum on each axis (in the spacecraft reference frame) at the end of a momentum dump (kilogram-meter^2/second)	Real	3	24	TRW PCAD	4.1.3.13.3 (Predict Spacecraft Momentum Dump Duration)	ω_f
55C	odb_max_momdist	Defines the maximum rate of momentum build-up on each axis in the spacecraft reference frame (kilogram-meter^2/second^2)	Real	3	24	TRW PCAD	N/A	
	ODE_SI_Constraints		**Fixed**		**80**			
56	odb_max_rad	Defines maximum allowable radiation flux for a science instrument. (particles/cm*2; index: 1 = electron and proton, 2=ACIS-I, ACIS-S, HRC-I, HRC-S.	Real	2x4	64	ASC	N/A[2]	
57	odb_opt_rad	Defines maximum radiation energy for a science instrument (mev; index: 1 = electron and proton, 2 = ACIS-I, ACIS-S, HRC-I, HRC-S.	Real	2x4	64	ASC	N/A[2]	
58	odb_isi	Defines which sis are usable, index: ACIS-I, ACIS-S, HRC-I, HRC-S	Logical	4	16	TBD	4.1.5 (support SI configuration)	N/A
59	odb_grating	Defines which gratings are usable, index: HETG, LETG	Logical	2	8	TBD	4.1.5 (support SI configuration)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
59A	odb_max_tsc_con	Defines the maximum continuous ON time for SIM TSC mechanism motion (seconds)	Real		8	TRW	4.1.5 (support SI configuration)	N/A
59B	odb_max_fa_con	Defines the maximum continous ON time for SIM FA mechanism motion (seconds)	Real		8	TRW	4.1.5 (support SI configuration)	N/A
59C	odb_tsc_duty_cycle	Defines the maximum duty cycle for SIM TSC mechanism motion (seconds; index: 1=maximum ON duration, 2=time interval, 3=maximum recovery interval)	Real	3	24	TRW	4.1.5 (support SI configuration)	$\tau_{TSC,ON}$, $\tau_{TSC,int}$, $\tau_{TSC,r}$
59D	odb_fa_duty_cycle	Defines the maximum duty cycle for SIM FA mechanism motion (seconds; index: 1=maximum ON duration, 2=time interval, 3=maximum recovery interval)	Real	3	24	TRW	4.1.5 (support SI configuration)	$\tau_{FA,ON}$, $\tau_{FA,int}$, $\tau_{FA,r}$
	ODE_Power_Constraints		**Fixed**		**80**			
60	odb_max_dod	Defines maximum depth of discharge.	Real		8	TRW EPS	N/A[2]	
61	odb_max_load	Defines maximum spacecraft load. (watts)	Real		8	TRW EPS	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
62	odb_min_shdw_gap	Defines minimum recovery tome between successive Earth and Lunar shadow events. Shadow events closer together than this time will be merged (seconds)	Relative Time		8	TRW SE	N/A[2]	
	ODE_LGA_Constraints		**Fixed**		**80**			
63	odb_lga_sa	Defines line of sight avoidance for blockage of the LGA by the solar arrays (meters: index: 1=LGA, 2=solar array)	Real	2x2	32	TRW MDI	4.1.8.3 (Calculate SA blockage)	ε
64	odb_lga_ssh	Defines line of sight avoidance for blockage of the LGA by the sunshade door (meters; index: LGA)	Real	2	16	TRW MDI	4.1.8.2 (Calculate Sunshade Door Blockage)	ε_L
65	odb_lga_body	Defines line of sight avoidance for blockage of the LGA by the spacecraft body (meters; index: LGA)	Real	2	16	TRW MDI	4.1.8.4 (Calculate S/C body blockage)	ε_L
66	odb_ilga	Defines LGA usability indicator (index: LGA ID) (=.TRUE., Hardware usable)	Logical	2	8	TRW SE	4.1.8.5 (select optimal DSN station and LGA)	N/A
67	odb_ssopen	Defines sunshade door open indicator (=.TRUE., OPEN)	Logical		4	TRW SE	4.1.8.2 (Calculate Sunshade Door Blockage)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
	ODE_Recorder_Constraints		**Fixed**		**80**			
68	odb_ssr	Defines SSR hardware usability indicator (.TRUE. = SSR hardware usable; index: SSR)	Logical	2	8	TRW SE	4.1.9.1, 4.1.9.2 (calculate current recorder usage, check playback support requirements)	N/A
	ODE_SA_Constraints		**Fixed**		**80**			
69	odb_isa1	Defines SA usability indicator, =.TRUE, SA 1 is usable	Logical		4	TRW SE	4.4.1.12 (compute SA power)	N/A
70	odb_isa2	Defines SA usability indicator, =.TRUE, SA 2 is usable	Logical		4	TRW SE	4.4.1.12 (compute SA power)	N/A
71	odb_sa1max	Defines maximum allowed rotation angle from the AXAF body frame x-axis to the SA-1 normal in the AXAF body frame x-z plane (about the AXAF body frame y-axis) (degrees)	Real		8	TRW PCAD	4.1.3.3 (calculate solar array rotation angles)	θ_u
72	odb_sa1min	Defines minimum allowed rotation angle from the AXAF body frame x-axis to the SA-1 normal in the AXAF body frame x-z plane (about the AXAF body frame y-axis) (degrees)	Real		8	TRW PCAD	4.1.3.3 (calculate solar array rotation angles)	θ_l

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
73	odb_sa2max	Defines maximum allowed rotation angle from the AXAF body frame x-axis to the SA-2 normal in the AXAF body frame x-z plane (about the AXAF body frame y-axis) (degrees)	Real		8	TRW PCAD	4.1.3.3 (calculate solar array rotation angles)	θ_u
74	odb_sa2min	Defines minimum allowed rotation angle from the AXAF body frame x-axis to the SA-2 normal in the AXAF body frame x-z plane (about the AXAF body frame y-axis) (degrees)	Real		8	TRW PCAD	4.1.3.3 (calculate solar array rotation angles)	θ_l
74A	odb_max_salsun	Defines the maximum allowed angle between the solar array 1 normal and the Sun during nudge mode operations.	Real		8	TRW SE	N/A	
74B	odb_max_sassun	Defines the maximum allowed angle between the solar array 2 normal and the Sun during nudge mode operations.	Real		8	TRW SE	N/A	
	ODE_Command_Constraints		**Fixed**		**80**			
75	odb_num_pairs	The number of command pair timing constraints specified in odb_cmd_pair, up to 1000	Integer		4	TRW CCDM	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
76	odb_cmd_pair	Command pairs with specific timing constraints (index: 1 = command pair timing information, - first command in pair (mnemonic) - second command in pair (mnemonic) - minimum time separation between the command Pair (relative time Index: 2 = command pair)	ASCII	3x1000	60000	TRW CCDM	N/A[2]	
77	odb_cmd_per_sec	Maximum number of stored commands allowed per second	Integer		4	TRW CCDM	N/A[2]	
78	odb_rctu_cmd_sp	Minimum time spacing between of a given type to an RCTU (seconds; index: 1 = command type (HLD, LLD); 2 = RCTU)	Real	2x5	80	TRW CCDM	N/A[2]	
79	odb_rctu_cmd_mask	Bit mask used to identify commands of a given type for a given RCTU (index: 1 = command type (HLD, LLD); 2 = RCTU	Integer	2x5	40	TRW CCDM	N/A[2]	
80	odb_max_daily_ld	Maximum size of the daily stored command load (words)	Integer		4	TRW CCDM	4.1.7 (Estimate OBC Memory Usage)	N/A
81	odb_max_onboard_ld	Maximum size of the nominal onboard stored command loads (words)	Integer		4	TRW CCDM	4.1.7 (Estimate OBC Memory Usage)	N/A
82	odb_max_stored_cmds	Maximum size for all onboard absolute time stored command loads	Integer		4	TRW CCDM	4.1.7 (Estimate OBC Memory Usage)	N/A

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
83	odb_ats_cmd_consts	Command mnemonics that cannot be referenced from a command sequence definition file	ASCII	350	4000	TRW CCDM		
84	odb_rts_cmd_consts	Command mnemonics that cannot be referenced from a relative time sequence definition file	ASCII	350	4000	TRW CCDM		
85	odb_obc_id	Identifies the currently on-line (primary) OBC; allowed values are “A” or “B”	ASCII		1	TRW CCDM	N/A[2]	
85A	odb_editstar_sep	Defines the minimum time separation between successive Edit Star Catalog commands (seconds)	Real		8	TRW PCAD	4.1.11.1.2 (Specify an Acquisition)	N/A
85B	odb_obccmds_mc	Defines the maximum number of commands executing from ground-controlled SCSs per minor cycle; index = OBC software function (1 = Executive, 2 = CCDM, 3 = Health and Status, 4 = PCAD, 5 = EPS, 6 = Telescope Support)	Integer	6	24	TRW FSW	N/A	
86	odb_num_slice_cmd	Number of slice commands defined	Integer		4	TRW CCDM	N/A[2]	
87	odb_slice_cmd_sp	Minimum time spacing between slice commands (seconds; indes:slice command)	Real	10	80	TRW CCDM	N/A[2]	

No.	*Record identifier/ variable name*	*Description*	*Record Format/ Field Format*	*Dimen*	*Record Length/ Field Length (Bytes)*	*Value Provider*	*OFLS SDS Section Number*	*Variable Name*
	ODE_Maneuver_Constraints		**Fixed**		**80**			
88	odb_slice_cmd_mask	Bit mask used to identify slice command type (index slice command type (index; slice command)	Integer	10	40	TRW CCDM	N/A[2]	
89	odb_min_one_bits	Defines the number of "1" bits to trigger insertion of a CTU NOOP command. If the 28 data bits of a command has less than this number of 1 bits, CTU NOOP commands will be inserted in OFLS-generated command load data files.	Integer		4	TRW CCDM	N/A	

3.11 DSN APPROVED SCHEDULES

3.11.1 HEADER INFORMATION

Element Type:	ODE
Format provided by:	JPL
Data provided by:	JPL
Data Routinely Accessed by:	OFLS (MPS, CM)
Format Structure:	ASCII Flat File
Delivery Media:	Electronic File Transfer

Mission Planning DSN schedule files are prepared by the FOT and placed in the ODE using the naming convention DSN**sss_eee**.mp**X** where,

sss is the starting day of the DSN schedule in TJD

eee is the ending day of the DSN schedule in TJD

X is a revision letter (a, b, c….)

3.11.2 CONTENT

This element contains data defining the scheduled and approved CXO DSN contacts to be used in mission planning for each schedule period. DSN schedule files are prepared by the FOT based on data extracted from the current Deep Space Network (DSN) published communications schedules converted to an OFLS MPS compatible format..

3.11.3 FORMAT DESCRIPTION

Reference MSFC to JPL ICD, Appendix G.

3.12 DSN SCHEDULE REQUESTS

3.12.1 HEADER INFORMATION

Element Type:	ODE
Format provided by:	JPL
Data provided by:	OFLS (MPS)
Data Routinely Accessed by:	OFLS (MPS)

Format Structure:	ASCII Flat File
Delivery Media:	Electronic File Transfer

3.12.2 CONTENT

This element contains parameters for schedule request including: project ID, antenna number, start and end time of activity, etc.

3.12.3 FORMAT DESCRIPTION

Reference MSFC to JPL ICD, Appendix G.

3.13 ENGINEERING REQUEST

3.13.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	TRW, ASC
Data Routinely Accessed By:	OFLS (MPS)
Format Structure:	ASCII, Sequential
Delivery Media:	Electronic Transfer

3.13.2 CONTENT

The engineering request (ER) list contains individual statements for each requested engineering activity. ERs are generated by the FOT and the ASC and are used as inputs to the mission schedule generation process. The types of ERs that will be made include:

momentum dump

solid state recorder dump

OBC dump

OBC reconfiguration (e.g. test a patch on backup OBC)

5 special communications support (e.g. requested by DSN to test new sites, software, etc.)

sensor performance investigation (calibration, performance characterization)

special preparations for eclipse related events

power/thermal configuration and maintenance.

The specific content of each ER will depend upon the type of request. The ER List can have any name, but must end with the extension .ER.

3.13.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 50,000 (Bytes)

RECORD LAYOUT:

Record Identifier: ER Request Record

Record Format: Fixed

Record Length: 80 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_erreq_line	Defines one line of ER request input.	ASCII		80

3.13.3.1 ER Request Syntax Generic Information

The following pages provide the format and contents of ERs. Each ER is a separate statement. Currently, the statements allowed in an ER list are CAL, HDR, BEGIN_COMMENT, END_COMMENT, MOM, LINK, PBK, COMM, TLM, and ACT. The HDR, BEGIN_COMMENT, and END_COMMENT are identical to the OR list HDR, BEGIN_COMMENT, and END_COMMENT statements. All the statements are described below. ER statements follow the same conventions described in Section 3.24 for OR lists.

3.13.3.2 CAL Statement Syntax

CAL, ID=string, [TARGET=(ra, dec, name)|MANEUVER = (v1, v2, v3, angle, ref), |SS_OBJECT=string,][SLEW_HW=string], [SL_RATE=real]DURATION=(nominal,min_duration, max_duration), PRIORITY=Integer[,SI=ACIS-I|ACIS-S|HRC-I |HRC-S|ESA|FSS], [,STAR=(ra,dec,mag,type,id)], MIN_GUIDE, MIN_ACQ [,FID=(fidid, mag)] [,GRATING=HETG|LETG |NONE] [,SI_MODE=mnemonic] [,BIAS=mnemonic] [,ACA_MODE=mnemonic] [,CLASS=string] [,ROLL=(roll_angle,roll_tolerance)] [,TARGET_OFFSET=(y_offset, z_offset)] [,SIM_OFFSET=(trans_offset, focus_offset)] [,DITHER=(status, y_amp, y_freq, y_phase , z_amp, z_freq, z_phase)] [,WINDOW=(window_start_time, window_end_time)] [,PHASE=(period, epoch, start_range, start _margin, end_range, end_margin)] [,REPETITION=(period, delta, number_reps)] [,PRECEDING=(reqid, minimum_lead, maximum_lead)] [,SEGMENT=(max_number, min_duration, max_separation)], [MOON=(status, avoid_angle)] [,SUN=(status, avoid_angle)] **[,**FSS=(status, fss_fov_angle**)]** [EARTH=(status, avoid_angle)] [,PLANET=(id, status, avoid_angle)] [,OBJECT=(id, status, avoid_angle)] , [,E_RADIATION=(status, energy, flux)][, P_RADIATION=(status,energy, flux)] [,ECLIPSE=(status,DAY|NIGHT)] [,OVERLAP=(reqid, start_lead, end_lead)] [,ALTITUDE=(status, min_altitude, max_altitude)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	ID	Request identifier	ASCII string	n/a	1-8 char	n/a
	{ **TARGET**					
*	ra	Target position right ascension	Real	degrees	0.0 - 360.0	TBD
*	dec	Target position declination	Real	degrees	-90.0 - +90.0	TBD
	name	Target name	Unrestricted string	n/a	1-20 char	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	{**MANEUVER**					
*	v1	Maneuver unit eigenvector component 1	Real	n/a	0 - 1	TBD
*	v2	Maneuver unit eigenvector component 2	Real	n/a	0 - 1	TBD
*	v3	Maneuver unit eigenvector component 3	Real	n/a	0 - 1	TBD
*	angle	Maneuver angle	Real	degrees	0.0 - 360.0	TBD
	ref	Reference frame the eigen vector is defined in	ASCII	n/a	GCI/SC	n/a
	SS_OBJECT MT_OBJECT}	Name of solar system object to view(defined in the solar, lunar, planetary ephemeris) Name of solar system object to view (defined in the ODE-Object Characteristics) to view.	ASCII string ASCII	n/a n/a	EARTH| MOON| MARS|ME RCURY| VENUS| JUPITER| SATURN| URANUS| NEPTUN E| PLUTO| SUN 1-20 Char	n/a n/a
	SLEW_HW	Indicates how slew is performed (RCS or RWA)	string	n/a	RCS|RWA	n/a
	SL_RATE	Defines the rate at which the slew to the target is to be performed	Real	degrees/second	n/a	n/a
*	**DURATION**					
*	nominal	Request duration	Real	seconds	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	min_duration	Minimum request duration	Real	seconds	n/a	n/a
	max_duration	Maximum request duration	Real	seconds	n/a	n/a
*	PRIORITY	Request priority (highest to lowest)	Integer	n/a	1 - 10	n/a
	SI	Request science instrument or spacecraft sensor	ASCII string	n/a	ACIS-I | ACIS-S | HRC-I | HRC-S | ESA | FSS	n/a
	STAR					
*	ra	Star right ascension	Real	degrees	0.0 - 360.0	0.0001
*	dec	Star declination	Real	degrees	-90.0 - +90.0	0.0001
*	mag	Star instrumental magnitude	Real	mag	-10.0 - +20.0	0.01
	type	Star type: guide, acquisition, or both guide and acquisition	ASCII	n/a	GST| ACQ|BOTH	n/a
	id	Star catalog id	Integer	n/a	-1e8 to +2e9	n/a
	MIN_GUIDE	Minimum number of guide stars to be selected	Integer	n/a	0 - 8	n/a
	MIN_ACQ	Minimum number of acquisition stars to be selected	Integer	n/a	0 - 8	n/a
*	fidid	FID light identifier	Integer	n/a	1 - 14	n/a
	mag	FID light instrumental magnitude	Real	mag	-10.0 - +20.0	0.01
	GRATING	Request grating	ASCII string	n/a	LETG|HETG |NONE	n/a
	SI_MODE	Science request mode	Mnemonic	n/a	1-10 char	n/a
	BIAS	ACIS Bias Option	Mnemonic	n/a	OPT|REQ| NONE	n/a
	ACA_MODE	ACA request mode	Mnemonic	n/a	1-8 char None	n/a
	CLASS	Request class	ASCII string	n/a	TBD	n/a

AX000600
SCN 014

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	ROLL					
*	roll_angle	Target roll	Real	degrees	0 - 360.0	n/a
	roll_tolerance	Target roll tolerance	Real	degrees	n/a	n/a
	TARGET_OFFSET					
	y_offset	Target position offset in the y-direction (rotation around the z-axis)	Real	degrees	-1.0 - +1.0	n/a
	z_offset	Target position offset in the z-direction (rotation around the y-axis)	Real	degrees	-1.0 to +1.0	n/a
	SIM_OFFSET					
	trans_offset	Offset from nominal of translation position.	Real	SIM translation motor steps	HRC-S: -4989 ~ +203807 HRC-I: -53791 ~ +203807 ACIS-S: -179917 ~ +28879 ACIS-I: -197869 ~ +10927	SIM Motor Step
	focus_offset	Offset from nominal of focus position.	Real	SIM focus motor steps	-11416 ~ +6562	SIM Motor Step
	DITHER					
	status	Status of onboard dither processing	Mnemonic	n/a	ON|OFF	n/a
	y_amp	Spacecraft dither amplitude in the y-axis direction of the spacecraft body frame	Real	deg	0.0000 ~ +5.555E-3	0.0001
	y_freq	Spacecraft dither frequency in the y-axis direction of the spacecraft body frame	Real	deg/sec	0.0001 ~ +1.5758	0.0001
	y_phase	Spacecraft dither phase in the y-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +360.0000	0.0001

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	z_amp	Spacecraft dither amplitude in the z-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +5.555E-3	0.0001
	z_freq	Spacecraft dither frequency in the z-axis direction of the spacecraft body frame	Real	deg/sec	0.0001 ~ +1.5758	0.0001
	z_phase	Spacecraft dither phase in the z-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +360.0000	0.0001
	WINDOW					
	window_start_time	Earliest requested start time	Absolute time	GMT	n/a	n/a
	window_end_time	Latest requested end time	Absolute time	GMT	n/a	n/a
	PHASE					
*	period	Object ephemeris period	Relative time	GMT	n/a	n/a
*	epoch	Absolute time for phase 0	Absolute time	GMT	n/a	n/a
	start_range	Start range of phase to be observed	Real	phase	0 - 1	n/a
	start_margin	Allowed range in phase interval starting point	Real	phase	0 - 1	n/a
	end_range	End range of phase to be observed	Real	phase	0 - 1	n/a
	end_margin	Allowed range in phase interval ending point	Real	phase	0 - 1	n/a
	REPETITION					
*	period	Repetition interval	Relative time	GMT	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	GMT	n/a	n/a
	number_reps	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
*	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	minimum_lead	Minimum time between requests	Relative time	GMT	n/a	
	maximum_lead	Maximum time between requests	Relative time	GMT	n/a	n/a
	SEGMENT					
	max_number	Maximum number of segments	Integer	n/a	>0	n/a
	min_duration	Minimum duration of any segment	Real	Seconds	n/a	n/a
	max_separation	Maximum separation between segments	Real	Seconds	n/a	n/a
	MOON					
*	status	Status of Moon occultation check	ASCII string	n/a	ON|OFF	n/a
	avoid_angle	Moon occultation check angle	Real	degrees	0.0 - 360.0	TBD
	SUN					
*	status	Status of Sun occultation check	ASCII string	n/a	ON|OFF	n/a
	avoid_angle	Sun occultation check angle	Real	degrees	0.0 - 360.0	TBD
	FSS					
	status	Status of FSS FOV check	ASCII string	n/a	ON|OFF	n/a
	fss_fov_angle	FSS FOV check angle	Real	degrees	0.0 - 360.0	n/a
	EARTH					
*	status	Status of earth occultation check	ASCII string	n/a	ON|OFF	n/a
	avoid_angle	Earth occultation check angle	Real	degrees	0.0 - 360.0	TBD
	PLANET					
*	id	Planet identification	Mnemonic	n/a	1-8 char	n/a
*	status	Status of planet occultation check	ASCII string	n/a	ON|OFF	n/a
	avoid_angle	Planet occultation check angle	Real	degrees	0.0 - 360.0	TBD
	OBJECT					

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	id	X-Ray object identification	Mnemonic	n/a	1-8 char	n/a
*	status	Status of object occultation check	ASCII string	n/a	ON|OFF	n/a
	avoid_angle	Object check angle	Real	degrees	0.0 - 360.0	TBD
	E_RADIATION					
*	status	Status of Electron Radiation Check	ASCII string	n/a	ON|OFF	n/a
	energy	Electron Energy Level	Real	MeV	n/a	n/a
	flux	Electron Flux Level	Real	con^2sec	n/a	n/a
	P_RADIATION					
*	status	Status of Proton Radiation Check	ASCII string	n/a	ON|OFF	n/a
	energy	Proton Energy Level	Real	MeV	n/a	n/a
	flux	Proton Flux Level	Real	con^2sec	n/a	n/a
	ECLIPSE					
*	status	Status of Eclipse (Day/Night) Check	ASCII string	n/a	ON|OFF	n/a
	day/night	Indicates Desired Eclipse Condition for Scheduling Observation	ASCII string	n/a	DAY|NIGHT	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a
	ALTITUDE					
	status	Status of altitude check	mnemonic	n/a	ON|OFF	n/a
*	min_altitude	Minimum altitude at which request can be scheduled	Real	km	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	max_altitude	Maximum altitude at which request can be scheduled	Real	km	n/a	n/a

CAL

The calibration statement is used to request specific science or engineering calibrations and specify the mode of operation and special constraints associated with the request. Calibrations can be specified in terms of a specific target location on the sky or solar system object and/or in terms of maneuver across a portion of the sky. The allowed CAL parameter definitions are as follows:

ID — The ID parameter provides a unique identifier that is retained and passed to command management for tracing each request through ground processing, onboard execution, and post-request data processing. This is a required parameter.

TARGET — The TARGET parameter is used to specify the target location for a request. The TARGET parameter has three arguments: ra is the right ascension in J2000 coordinates of the requested target, dec is the declination in J2000 coordinates of the requested target, name is the object name of the requested target. The sub-parameters ra and dec are required; the sub-parameter name is optional. If the SI parameter has not been provided, then either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each request.

Multiple TARGET parameters may be specified. However, TARGET parameters may not be mixed with MANEUVER and SS_OBJECT and MT_OBJECT parameters on a single observation statement. If multiple TARGET parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each TARGET parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide stars retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

MANEUVER — The MANEUVER parameter is used to specify a maneuver across an extended region of the sky. The MANEUVER parameter has five arguments: v1, v2, and v3 are the components of the unit vector specifying the eigenaxis of the maneuver; angle is the maneuver angle about the eigenaxis, ref identifies whether the eigenaxis is defined in the GCI or the spacecraft reference frame.

If the SI parameter has not been provided, then either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each request.

Multiple MANEUVER parameters may be specified. However, MANEUVER parameters may not be mixed with TARGET, SS_OBJECT and MT_OBJECT parameters on a single observation statement. If multiple MANEUVER parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each MANEUVER parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide starts retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

SS_OBJECT

The SS_OBJECT parameter is used to specify a solar system object (such as the Moon, Jupiter, etc.) as the view target. The SS_OBJECT must be defined in the SLP file. The object position computed from the SLP data is centered in the sensor FOV at the mid point of the observation. If the SI parameter has not been provided then either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each observation.

Multiple SS_OBJECT parameters maybe specified. However, SS_OBJECT parameters may not be mixed with MANEUVER, TARGET, and MT_OBJECT parameters on a single observation statement. If multiple SS_OBJECT parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each MANEUVER parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide starts retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

SLEW_HW

The SLEW_HW parameter specifies how the slew to the requested target is performed. The target can be specified using the TARGET, MANEUVER, or SS_OBJECT parameter. Allowed values are RCS or RWA. This is an optional parameter. If no value is provided, the slew is assumed to be performed using the RWA.

SL_RATE	The SL_RATE parameter specifies how fast the slew to the requested target is performed. The SL_RATE parameter can be used with either the TARGET, MANEUVER, or SS_OBJECT parameter. The value for the SL_RATE parameter is dependent on the value of the SLEW_HW parameter, if provided, and must be greater than 0 and less than or equal to the maximum slew rate specified in the AXAF Constraints. This is an optional parameter. If no value is provided, the rate is assumed to be the maximum allowed slew rate.
DURATION	The DURATION parameter specifies the requested duration for the request. The DURATION parameter has three arguments: nominal is the desired duration, min_duration is the minimum duration allowed, and max_duration is the maximum duration allowed. The request may be extended beyond the nominal duration in order to utilize spacecraft idle time during the scheduling process, however it may not extend past the maximum duration. The min_duration and max_duration subparameters are optional. If omitted, the request will be scheduled at the nominal duration with no adjustment. The DURATION parameter and the nominal subparameter are required.
PRIORITY	The PRIORITY parameter specifies the priority of this request for scheduling. The priority is an Integer number from 1 to 10 (1 indicates highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of request priority. This is a required parameter.
SI	The SI parameter specifies the requested science instrument or spacecraft sensor to be used for the request. When specifying the ACIS or HRC instruments, this parameter specifies use of the non-focal plane SI and neither the TARGET, MANEUVER, or SS_OBJECT parameters may be provided. Use of the focal plane instrument is requested through the OR List OBS statement. When specifying a spacecraft sensor (ESA or FSS), the TARGET, MANEUVER, or SS_OBJECT parameters must be provided. Allowed values are ACIS-I, ACIS-S, HRC-I, HRC-S, ESA, or FSS. The is an optional parameter.
MT_OBJECT	The MT_OBJECT parameter is used to specify a solar system object (such as an asteroid or a comet) as the viewing target. The MT_OBJECT must be defined in the ODE_Object_ Characteristics record of the Characteristics file. The object position computed from the orbital elements in the ODE is centered in the sensor FOV at the mid-point of the observation. If the SI parameter has not been provided, then either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each observation.

	Multiple MT_OBJECT parameters may be specified. However, MT_OBJECT parameters may not be mixed with MANEUVER, TARGET, and SS_OBJECT parameters of a single observation statement. If multiple MT_OBJECT parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each MANEUVER parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide starts retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).
STAR	The STAR parameter specifies a guide star or acquisition star. The star parameter has five arguments: ra is the right ascension in J2000 coordinates, dec is the declination in J2000 coordinates, mag is the instrumental magnitude of the star, type specifies whether the star is a guide star, an acquisition star, a monitor window, or is used for both acquisition and guide star determination, and id specifies the star catalog identifier for the star (if any). Multiple STAR parameters may be specified. The total number of STAR plus FID parameters may not exceed 13. This is an optional parameter.
FID	The FID parameter specifies a fid light. The FID parameter has two arguments: fidid is the fiducial light identifier and mag is the instrumental magnitude of the fid lights. Multiple FID parameters may be specified. The total number of STAR plus FID parameters may not exceed 13. This is an optional parameter.
MIN_GUIDE	The MIN_GUIDE parameter specifies the minimum number of guide stars to be selected for the current request. If the value for MIN_GUIDE is greater than 5, then the number of FID lights selected will be (8 - MIN_GUIDE). If guide stars are specified by the STAR parameter and MIN_GUIDE is specified, then the number of guide stars selected will be at least (MIN_GUIDE - number of guide stars specified). This parameter is optional. If MIN_GUIDE is not specified, the software will use the default value in the ODE Characteristics.
MIN_ACQ	The MIN_ACQ parameter specifies the minimum number of acquisition stars to be selected for the current request. If acquisition stars are specified by the STAR parameter and MIN_ACQ is specified, then the number of guide stars selected will be at least (MIN_ACQ - number of acquisition stars specified). This parameter is optional. If MIN_ACQ is not specified, the software will use the default value in the ODE Characteristics.

GRATING	The GRATING parameter specifies the transmission grating to be used for the request. Allowed values are HETG, LETG, or NONE. This is an optional parameter.
SI_MODE	The SI_MODE parameter specifies the operational mode for ACIS or HRC requests depending on the SI parameter. This parameter is a mnemonic used for a look-up into the table of modes to obtain the correct table parameters for the request. A default operational mode is provided using the mnemonic DEFAULT. The schedule generation software will maintain history on the operational mode and provide the table parameters for uplink to the spacecraft if the operational mode changes between consecutively scheduled requests. This is an optional parameter.
BIAS	The BIAS parameter specifies the bias measurement options for ACIS observations. Allowed values are OPT, REQ, or NONE. If OPT is specified, then the MPS determines if a bias is needed based on the scheduling rules. If REQ is specified, then a bias measurement will always be scheduled. If NONE is specified, then no bias measurement will be scheduled. This is an optional parameter. The BIAS parameter can only be specified if the value of the parameter SI is ACIS-I or ACIS-S.
ACA_MODE	The ACA_MODE parameter specifies the operational mode for ACA requests. This parameter is a mnemonic used for a look-up into the table of modes to obtain the correct table parameters for the observation. If the ACA_Mode parameter is not provided on the statement, a default mode of NONE is assumed and the calibration request is performed without ACA support.
CLASS	The CLASS parameter specifies the class of requests to which this request belongs. The CLASS parameter is used by certain scheduling algorithms to provide balanced coverage between the different types of requests. The allowed values are TBD. This is an optional parameter.
ROLL	The ROLL parameter specifies a specific spacecraft roll for the request. The ROLL parameter has two arguments: roll_angle is the position angle of the AXAF Z-axis projected onto the sky at the nominal boresight pointing of AXAF measured from North through East and roll_tolerance is the allowed tolerance of the roll angle. ROLL is specified in decimal degrees in celestial coordinates. Because of roll constraints to maintain proper Sun pointing, the roll parameter will constrain the time window within which the request can be scheduled. This is an optional parameter; a ROLL may only be supplied if the TARGET or SS_OBJECT parameter was supplied.

TARGET_OFFSET	The TARGET_OFFSET parameter specifies an offset between the spacecraft boresight and the target position specified by the TARGET parameter. The TARGET_OFFSET parameter has two arguments: y_offset is the offset along the y-axis (rotation around the z-axis in the SI reference frame), z_offset is the offset along the z-axis (rotation around the y-axis in the SI reference frame). A positive offset in either the y or z direction causes the spacecraft boresight to be offset negatively in the y or z direction, respectively. This is an optional parameter; a TARGET_OFFSET may only be supplied if the TARGET or SS_OBJECT parameter was supplied.
SIM_OFFSET	The SIM_OFFSET parameter specifies an offset from the nominal SIM position. The SIM_OFFSET parameter contains two arguments: trans_offset is the offset from nominal of translation position and focus_offset is the offset from nominal of focus position. This parameter is optional.
DITHER	The DITHER parameter specifies the dither mode for the request. The DITHER parameter has 7 arguments. Status indicates whether spacecraft dither is on or off for the observation. The six parameters: y_amp, y_freq, y_phase, z_amp, z_freq, z_phase, provide values for the spacecraft y-axis amplitude, frequency, and phase and the spacecraft z-axis amplitude, frequency, and phase, respectively. The schedule generation software will maintain history on the current commanded dither mode and values and provide the dither command parameters for uplink to the spacecraft if either the dither mode or the dither parameters change between consecutively schedule observations. This is an optional parameter. If the parameter is omitted, the observation will default to dither on using the nominal dither amplitude, frequency, and phase from the AXAF characteristics. If the status is specified as on and any or all of the six dither parameters are omitted, the values will default to the nominal values specified in the AXAF characteristics. If the status is specified as off, any additional dither parameters will be ignored.
WINDOW	The WINDOW parameter is used to provide absolute time constraints on the scheduling of the request. The WINDOW parameter has two arguments: window start time and window end time. The request must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the request by providing a window which is equal to the minimum duration of the request as specified in the DURATION parameter. Omitting the window end argument indicates that the observation can be scheduled at any time after the specified time. Omitting the window start argument indicates

	that the request must complete any time prior to the specified time. This is an optional parameter.
PHASE	The PHASE parameter specifies time constraints on the scheduling of the request based on the ephemeris of the observed object. The PHASE parameter has six arguments: period specifies the phase period of the object, epoch specifies the absolute time for phase 0, start_range specifies the start of the phase, end_range specifies the end of the phase, start_margin specifies an allowed tolerance in the starting phase for scheduling of the request, end_margin specifies an allowed tolerance in the ending phase for scheduling of the request. The PHASE parameter can be used in conjunction with the WINDOW parameter to constrain the overall time interval to which the ephemeris applies, and with the REPETITION parameter to specify the number of times the request is to be scheduled. This is an optional parameter. If the PHASE parameter is specified, only the period argument is required.
REPETITION	The REPETITION parameter is used to specify the number and interval for repetitive requests. The REPETITION parameter has three arguments: period specifies the period for the repetition of the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. If the REPETITION parameter is used in conjunction with the PHASE parameter, only the argument number can be supplied; the period and delta arguments will be ignored if provided. This is an optional parameter.
PRECEDING	The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the request ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
SEGMENT	The SEGMENT parameter specifies whether a request can be scheduled as separate segments and controls the segmentation of the request. The SEGMENT parameter has three arguments: max_number specifies the maximum number of segments in

	which the request can be scheduled, min_duration specifies the minimum duration of any segment, and max_separation specifies the maximum time separation between any two segments. This is an optional parameter.
MOON	The MOON parameter provides information to override the Moon occultation checks performed during the scheduling of the request. The MOON parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the Moon to be used in checking for Moon occultation and status is used to turn off the Moon occultation check completely. This parameter is optional. If Moon is provided, only the avoid_angle argument is required. If the MOON parameter or the status sub-parameter is omitted, the status of the Moon occultation check is assumed to be "on" and the occultation check will be performed.
SUN	The SUN parameter provides information to override the Sun occultation checks performed during the scheduling of the request. The SUN parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the Moon to be used in checking for Sun occultation and status is used to turn off the Sun occultation check completely. This parameter is optional. If Sun is provided, only the avoid_angle argument is required. If the SUN parameter or the status sub-parameter is omitted, the status of the Sun occultation check is assumed to be "on" and the occultation check will be performed.
FSS	The FSS parameter provides information to override the FSS FOV checks performed during the scheduling of the request. The FSS parameter has two arguments: fss_fov_angle specifies the FSS FOV angle in degrees from the boresight of the FSS to be used in checking if the Sun is in the FSS FOV for the requested target, and status is used to turn off the FSS FOV check completely. This parameter is optional. If FSS is provided, only the fss_fov_angle argument is required. If the FSS parameter or the status sub-parameter is omitted, the status of the FSS FOV check is assumed to be "on" and the FSS FOV check will be performed
EARTH	The EARTH parameter provides information to override the earth occultation checks performed during the scheduling of the request. The EARTH parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the earth to be used in checking for earth occultation and status is used to turn off the earth occultation check completely. This parameter is optional. If EARTH is provided, only the

	avoid_angle argument is required. If the EARTH parameter or the status sub-parameter is omitted, the status of the earth occultation check is assumed to be "on" and the occultation check will be performed.
PLANET	The PLANET parameter provides information to modify the planet occultation checks performed during the scheduling of the request. The PLANET parameter has three arguments: id specifies the planet occultation check to be modified, avoid_angle specifies the avoidance angle in degrees from the planet to be used in checking for planet occultation, and status is used to turn off the planet occultation check completely. This parameter is optional. If PLANET is provided, only the id and avoid_angle arguments are required. If the PLANET parameter or the status sub-parameter is omitted, the status of the planet occultation check is assumed to be "on" and the occultation check will be performed.
OBJECT	The OBJECT parameter provides information to modify the celestial object checks performed during the scheduling of the request. The OBJECT parameter has three arguments: id specifies the celestial object check to be modified and is used as a look-up into the ODB table of celestial object constraints, avoid_angle specifies the avoidance angle in degrees from the celestial object to be used in checking for celestial object interference, and status is used to turn off the celestial object occultation check completely. This parameter is optional. If OBJECT is provided, only the id and avoid_angle arguments are required. If the OBJECT parameter or the status sub-parameter is omitted, the status of the celestial object check is assumed to be "on" and the check will be performed.
E_RADIATION	The E_RADIATION parameter provides information to modify the radiation zone avoidance checks for electrons performed during the scheduling of the observation. The E_RADIATION parameter has three arguments: status is used to turn off the radiation zone avoidance check for electrons completely, energy is the electron particle energy, and flux is the electron particle flux level above which the observation cannot be scheduled. This parameter is optional. If E_RADIATION is provided, only the energy and flux parameters are required. If the E_RADIATION parameter or the status sub-parameter is omitted, the status of the electron radiation zone avoidance check is assumed to be "on" and the check will be performed.
P_RADIATION	The P_RADIATION parameter provides information to modify the radiation zone avoidance checks for protons performed during the scheduling of the observation. The P_RADIATION parameter has

	three arguments: status is used to turn off the radiation zone avoidance check for protons completely, energy is the proton particle energy, and flux is the proton particle flux level above which the observation cannot be scheduled. This parameter is optional. If P_RADIATION is provided, only the energy and flux parameters are required. If the P_RADIATION parameter or the status sub-parameter is omitted, the status of the proton radiation zone avoidance check is assumed to be "on" and the check will be performed.
ECLIPSE	The ECLIPSE parameter provides information to control scheduling of the observation during orbit day or orbit night. The ECLIPSE parameter has two arguments: status is used to turn off the spacecraft eclipse check completely, day/night is used to indicate during which portion of the orbit the observation must be scheduled. This parameter is optional. If ECLIPSE is provided, only the day/night parameter is required. If the ECLIPSE parameter or the status sub-parameter is omitted, the status of the eclipse check is assumed to be "on" and the observation will be scheduled in orbit day only.
OVERLAP	The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specifies the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
ALTITUDE	The ALTITUDE parameter specifies an altitude constraint check to be used during scheduling of the request. The ALTITUDE parameter has three arguments: status is used to turn on the altitude check, min_altitude and max_altitude specify the lowest and highest altitudes at which the request can be scheduled, respectively. This parameter is optional. If ALTITUDE is provided with a status of "on", either the min_altitude or max_altitude subparameters are required. If the ALTITUDE parameter or the status subparameter is omitted, the status of the altitude constraint check is assumed to be "off" and the check will not be performed.

3.13.3.3 MOM Statement Syntax

MOM, MAX_MOM=real | MAX_TIME=relative time,
PRIORITY=Integer[,WINDOW=(window_start_time, window_end_time)] ,ID=string
[,REPETITION=(period,delta,number)]
[,PRECEDING=(reqid,minimum_lead,maximum_lead)][,OVERLAP=(requid, start_lead, end_lead)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	MAX_MOM	Maximum allowed spacecraft angular momentum	Real	Kg-m^2/s	TBD	TBD
*	MAX_TIME }	Maximum time allowed between momentum dumps	Relative time	n/a	n/a	n/a
*	PRIORITY	Request priority	Integer	n/a	1 - 10	n/a
	WINDOW					
	window_start_time	Earliest requested start time	Absolute time	GMT	n/a	n/a
	window_end_time	Latest requested end time	Absolute time	GMT	n/a	n/a
*	ID	Request identifier	ASCII string	n/a	1-8 char	n/a
	REPETITION					
	period	Repetition interval	Relative time	n/a	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	n/a	n/a	n/a
	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char)	n/a
	minimum_lead	Minimum time between requests	Relative time	n/a	n/a	n/a
	maximum_lead	Maximum time between requests	Relative time	n/a	n/a	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a

MAX_MOM The MAX_MOM parameter specifies the maximum magnitude of the spacecraft momentum vector in the spacecraft body frame. Either MAX_MOM or MAX_TIME is required for each MOM statement.

MAX_TIME — The MAX_TIME parameter specifies the maximum allowed time between momentum dumps. Either MAX_MOM or MAX_TIME is required for each MOM statement.

PRIORITY — The PRIORITY parameter specifies the priority of this request for scheduling. The priority is an Integer number from 1 to 10 (1 indicates the highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of request priority. This is a required parameter.

WINDOW — The WINDOW parameter is used to provide absolute time constraints on the scheduling of the communications contact. The WINDOW parameter has two arguments: window start time and window end time. The request must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the request by providing a window which is equal to the minimum duration of the request as specified in the DURATION parameter. The WINDOW parameter can also be used to partially time constrain the scheduling of the observation. Specifying window start and end times that are less than the observation duration, indicates that the observation must cover the specified interval but places no restrictions on when the interval must occur during the observation. Omitting the window end argument indicates that the request can be scheduled at any time after the specified time. Omitting the window start argument indicates that the request must be completed any time prior to the specified time. This is an optional parameter.

ID — The ID parameter provides a unique identifier for tracing each request through ground processing and onboard execution. This is a required parameter.

REPETITION — The REPETITION parameter is used to specify the number and interval for repetitive requests. The REPETITION parameter has three arguments: period specifies the period for the repetition of the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. This is an optional parameter.

PRECEDING — The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.

OVERLAP — The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specified the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.

3.13.3.4 COMM Statement Syntax

COMM, LINK=FORWARD|RETURN|TRACKING,
DURATION=(nominal,min_duration,max_duration), PRIORITY=integer
[,WINDOW=(window_start_time,window_end_time)][,REPETITION=(period,delta,number)]
[,PRECEDING=(mnemonic,minimum_lead, minimum_lag)] [,DSN_STATION=
station_id|ANY], ID=mnemonic[,RATE=integer] [,OVERLAP=(reqid, start_lead, end_lead)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**LINK**	Communications link type	ASCII string	n/a	FORWARD | RETURN| TRACKING	n/a
*	**DURATION**					

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	nominal	Request duration	Real	seconds	n/a	
	min_duration	Minimum request duration	Real	seconds	n/a	n/a
	max_duration	Maximum request duration	Real	seconds	n/a	n/a
*	**PRIORITY**	Request priority	Integer	n/a	1 - 10	n/a
	WINDOW					
	window_start_time	Earliest requested start time	Absolute time	n/a	n/a	n/a
	window_end_time	Latest requested end time	Absolute time	n/a	n/a	n/a
	REPETITION					
	period	Repetition interval	Relative time	n/a	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	n/a	n/a	n/a
	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a
	minimum_lead	Minimum time between requests	Relative time	n/a	n/a	n/a
	maximum_lead	Maximum time between requests	Relative time	n/a	n/a	n/a
	DSN_STATION	Requested DSN ground station	ASCII string	n/a	1-6 char| ANY	n/a
*	**ID**	Request identifier	ASCII string	n/a	1-8 char	n/a
	RATE	Requested rate for the communications contact	integer	kbps	512/1024	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a

The communications statement is used to specify requirements for uplink, downlink, and tracking support through the DSN. The allowed COMM parameter definitions are as follows:

LINK	The LINK parameter specifies the type of communications link requested. The LINK parameter may have the value FORWARD, RETURN, or TRACKING. This is a required parameter.
DURATION	The DURATION parameter specifies the requested duration for the communications contact. The DURATION parameter has three arguments: nominal is the desired duration, min_duration is the minimum duration allowed, and max_duration is the maximum duration allowed. The request may be extended beyond the nominal duration in order to utilize spacecraft idle time during the scheduling process, however it may not extend past the maximum duration. The min_duration and max_duration subparameters are optional. If omitted, the request will be scheduled at the nominal duration with no adjustment. The DURATION parameter and the nominal subparameter are required.
PRIORITY	The PRIORITY parameter specifies the priority of this request for scheduling. The priority is an Integer number from 1 to 10 (1 indicates highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of request priority. This is a required parameter.
WINDOW	The WINDOW parameter is used to provide absolute time constraints on the scheduling of the request. The WINDOW parameter has two arguments: window start time and window end time. The request must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the request by providing a window which is equal to the minimum duration of the request as specified in the DURATION parameter. The WINDOW parameter can also be used to partially time constrain the scheduling of the observation. Specifying window start and end times that are less than the observation duration, indicates that the observation must cover the specified interval but places no restrictions on when the interval must occur during the observation. Omitting the window end argument indicates that the request can be scheduled at any time after the specified time. Omitting the window start argument indicates that the request must complete any time prior to the specified time. This is an optional parameter.
REPETITION	The REPETITION parameter is used to specify the number and interval for repetitive requests. The REPETITION parameter has three arguments: period specifies the period for the repetition of

	the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. This is an optional parameter.
PRECEDING	The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
DSN_STATION	The DSN_STATION parameter specifies the requested DSN ground station. The DSN_STATION parameter is an ASCII string with values equal to the valid station ids defined in the ODE or ANY if there is no restriction on selecting a DSN ground station. This is an optional parameter. If no DSN_STATION parameter is provided, the request defaults to scheduling on any ground station.
ID	The ID parameter provides a unique identifier for tracing each request through ground processing and onboard execution. This is a required parameter
RATE	The RATE parameter specifies the requested rate for the communications contact in kbps. This is an optional parameter. If no specified, the contact will default to the rate of the supporting contact in the approved DSN schedule.
OVERLAP	The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specifies the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.

3.13.3.5 PBK Statement Syntax

PBK, DURATION=(nominal,min_duration,max_duration), PRIORITY=Integer, ID=string, [,RATE=integer] [,WINDOW=(window_start_time,window_end_time)] [,REPETITION=(period,delta,number)] [,PRECEDING=(reqid, minimum_lead, maximum_lead)] [,OVERLAP=(reqid, start_lead, end_lead)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**DURATION**					
*	nominal	Request duration	Real	seconds	n/a	n/a
	min_duration	Minimum request duration	Real	seconds	n/a	n/a
	max_duration	Maximum request duration	Real	seconds	n/a	n/a
*	**PRIORITY**	Request priority	Integer	n/a	1 - 10	n/a
*	ID	Request identifier	ASCII string	n/a	1-8 char	n/a
	RATE	Requested playback data rate	Integer	kbps	512|1024	n/a
	WINDOW					
	window_start_time	Earliest requested start time	Absolute time	GMT	n/a	n/a
	window_end_time	Latest requested end time	Absolute time	GMT	n/a	n/a
	REPETITIONS					
	period	Repetition interval	Relative time	n/a	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	n/a	n/a	n/a
	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a
	minimum_lead	Minimum time between requests	Relative time	n/a	n/a	n/a
	maximum_lead	Maximum time between requests	Relative time	n/a	n/a	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a

PBK

The SSR playback statement is used to specify SSR playback requirements. The allowed PBK parameter definitions are as follows:

DURATION — The DURATION parameter specifies the requested duration for the playback contact. The DURATION parameter has three arguments: nominal is the desired duration, min_duration is the minimum duration allowed, and max_duration is the maximum duration allowed. The request may be extended beyond the nominal duration in order to utilize spacecraft idle time during the scheduling process, however it may not extend past the maximum duration. The actual expected duration of the playback will be computed based on the data stored on the SSR and reported in the command timeline. This is a required parameter

PRIORITY — The PRIORITY parameter specifies the priority of this request for scheduling. The priority is an integer number from 1 to 10 (1 indicates highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of the request priority. This is a required parameter.

ID — The ID parameter provides a unique identifier for tracing each request through ground processing and onboard execution. This is a required parameter

RATE — The RATE parameter is used to specify the playback data rate. This is an optional parameter. If no RATE parameter is provided, the request defaults to the high data rate, 1024.

WINDOW — The WINDOW parameter is used to provide absolute time constraints on the scheduling of the request. The WINDOW parameter has two arguments: window start time and window end time. The request must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the request by providing a window which is equal to the minimum duration of the request as specified in the DURATION parameter. The WINDOW parameter can also be used to partially time constrain the scheduling of the observation. Specifying window start and end times that are less than the observation duration, indicates that the observation must

	cover the specified interval but places no restrictions on when the interval must occur during the observation. Omitting the window end argument indicates that the request can be scheduled at any time after the specified time. Omitting the window start argument indicates that the request must be completed any time prior to the specified time. Either the WINDOW, REPETITION, PRECEDING, or OVERLAP parameter is required for each request.
REPETITION	The REPETITION parameter is used to specify the number and interval for repetitive requests. The REPETITION parameter has three arguments: period specifies the period for the repetition of the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. Either the WINDOW, REPETITION, PRECEDING, or OVERLAP parameter is required for each request.
PRECEDING	The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the request ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. Either the WINDOW, REPETITION, PRECEDING, or OVERLAP parameter is required for each request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same request id.
OVERLAP	The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specified the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. Either the WINDOW, REPETITION, PRECEDING, or OVERLAP parameter is required for each request. If both the PRECEDING and OVERLAP

parameters are specified, they cannot reference the same request id.

TLM Statement Syntax

TLM, FORMAT=mnemonic, PRIORITY=Integer, TIME=absolute_time,ID=string [,PRECEDING=(mnemonic,minimum lead,maximum lead)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**FORMAT**	Requested telemetry format	Integer	n/a	0-6	n/a
*	**PRIORITY**	Request priority	Integer	n/a	1 - 10	n/a
	TIME	Telemetry format change time	Absolute Time	n/a	n/a	n/a
*	**ID**	Request identifier	ASCII string	n/a	1-8 char	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a
	minimum lead	Minimum time between observations	Relative time	n/a	n/a	n/a
	maximum lead	Maximum time between observations	Relative time	n/a	n/a	n/a

TLM

The telemetry statement is used to request changes to the onboard telemetry format. The allowed TLM parameter definitions are as follows:

FORMAT — The FORMAT parameter specifies the requested telemetry format. The value is a mnemonic used as a look up to obtain the correct data rate for profiling SSR data storage. This is a required parameter.

TIME — The TIME parameter specifies the requested time for the telemetry format change. This is an optional parameter. However, if no TIME parameter is specified the request must reference another requests using the PRECEDING parameter or be reference by other request using the PRECEDING parameter.

PRIORITY — The PRIORITY parameter specifies the priority of this calibration observation for scheduling. The priority is an Integer number from 1 to 10 (1 indicates highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of calibration observation priority. This is a required parameter.

ID — The ID parameter provides a unique identifier for tracing each request through ground processing and onboard execution. This is a required parameter.

PRECEDING — The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests.

3.13.3.6 ACT Statement Syntax

ACT, SEQ=mnemonic, DURATION=(nominal[,min_duration][,max_duration]), PRIORITY=(integer) [,RESOURCE=(mnemonic)] [,WINDOW=(window_start_time, window_end_time)] [,REPETITION=(period,delta,number)] [,PRECEDING=(mnemonic,minimum_lead, maximum_lead)] ,ID=string [,pass_through] [,OVERLAP=(reqid, start_lead, end_lead)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**SEQ**	Command sequence definition	Mnemonic	n/a	n/a	n/a
*	**DURATION**					
*	nominal	Request duration	Real	seconds	n/a	n/a
	min_duration	Minimum request duration	Real	seconds	n/a	n/a
	max_duration	Maximum request duration	Real	seconds	n/a	n/a
*	**PRIORITY**	Request priority	Integer	n/a	1 - 10	n/a
	RESOURCE	Schedulable resource that will be used by activity	Mnemonic	n/a	ACIS, ACIS_NF, HRC, HRC_NF, HETG, LETG, FWD, RTN, PBK, TLM, MOM, MAN, ACA	
	WINDOW					
	window_start_ time	Earliest requested start time	Absolute time	GMT	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	window_end_time	Latest requested end time	Absolute time	GMT	n/a	n/a
	REPETITION					
	period	Repetition interval	Relative time	n/a	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	n/a	n/a	n/a
	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a
	minimum_lead	Minimum time between requests	Relative time	n/a	n/a	n/a
	maximum_lead	Maximum time between requests	Relative time	n/a	n/a	n/a
	ID	Activity identifier	ASCII string	n/a	1-8 char	n/a
	pass_through	Pass-through parameters	n/a	n/a	n/a	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a

ACT

The activity statement can be used to request an spacecraft activity to be scheduled that can be fully specified for scheduling by the OFLS MPS by a duration and a priority. By specifying the resource parameter, the ACT statement can also be used to set aside time for or preclude the use of a specific resource. The OFLS MPS will perform no processing for the resource used other than ensuring correct use of the compatibility rules as specified in the characteristics. The OFLS MPS will not calculate any output parameters except for the scheduled start and stop times. Spacecraft commanding will be based on the command sequence and pass-through parameters supplied in the request.

SEQ — The SEQ parameter specifies the command sequence definition to be used for this request. This parameter is a mnemonic passed to CM and used for a look-up into command sequence definition data element to determine the sequence of commands to perform this activity. This is a required parameter.

DURATION	The DURATION parameter specifies the requested duration for the observation. The DURATION parameter has three arguments: nominal is the desired duration, min_duration is the minimum duration allowed, and max_duration is the maximum duration allowed. The observation may be extended beyond the nominal duration in order to utilize spacecraft idle time during the scheduling process, however, it may not extend past the maximum duration. The min_duration and max_duration subparameters are optional. If omitted, the request will be scheduled at the nominal duration with no adjustment. The DURATION parameter and the nominal subparameters are required.
PRIORITY	The PRIORITY parameter specifies the priority of this activity for scheduling. The priority is an Integer number from 1 to 10 (1 indicates highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of request priority. This is a required parameter.
RESOURCE	The RESOURCE parameter specifies which, if any, schedulable resources are used to perform this activity. This parameter is used to ensure that the activity specified by the SEQ parameter will not be scheduled concurrently with other OR or ER request that utilize that resource. Multiple RESOURCE parameters can be used to specify each resource required to support the specified activity. ACIS indicates use of the ACIS in the focal plane, while ACIS_NF indicates use of the ACIS not in the focal plane. HRC indicates use of the HRC in the focal plane, while HRC_NF indicates use of the HRC not in the focal plane This is an optional parameter; omitting this parameter indicates that the activity can be scheduled concurrently with any other request.
WINDOW	The WINDOW parameter is used to provide absolute time constraints on the scheduling of the request. The WINDOW parameter has two arguments: window start time and window end time. The request must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the request by providing a window which is equal to the minimum duration of the request as specified in the DURATION parameter. The WINDOW parameter can also be used to partially time constrain the scheduling of the observation. Specifying window start and end times that are less than the observation duration, indicates that the observation must cover the specified interval but places no restrictions on when the interval must occur during the observation. Omitting the window end argument indicates that the request can be scheduled at any time after the specified time. Omitting the window start argument

	indicates that the request must be completed any time prior to the specified time. This is an optional parameter.
REPETITION	The REPETITION parameter is used to specify the number and interval for repetitive requests. The REPETITION parameter has three arguments: period specifies the period for the repetition of the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. This is an optional parameter.
PRECEDING	The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the ID for the request which must precede this request in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
ID	The ID parameter provides a unique identifier for tracing each activity through ground processing and onboard execution. This is an optional parameter. However, ID must be provided if this request is to referenced in the PRECEDING parameter of any other request.
pass_through	Pass through parameters may be provided on this statement in the form, parameter_name=value. If provided, the parameters will be included in the DOT statement for this request and used by CM as input values for building the commands in this request. The MPS will have no knowledge of the meaning of or utilize any of the pass-through parameters during scheduling.
OVERLAP	The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specifies the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. If both the PRECEDING and

OVERLAP parameters are specified, they cannot reference the same reqid.

3.13.3.7 SHDW Statement Syntax

SHDW,MIN_DUR=relative_time, MAX_DUR=relative_time[,ABSMIN_DUR=relative_time]
[,SA_OFFPT=string], TYPE=string [,WINDOW=(window_start_time,window_end_time)]
[,REPETITION=(period,delta,number)]
[,PRECEDING=(mnemonic,minimum_lead,maximum_lead)
[,OVERLAP=(reqid,start_lead,end_lead)][,ID=string[,pass_through]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	MIN_DUR	Minimum allowed shadow duration	Relative time	n/a	n/a	n/a
*	MAX_DUR	Minimum allowed shadow duration	Relative time	n/a	n/a	n/a
	ABSMIN_DUR	Absolute minimum duration for processed shadow events	Relative time	n/a	n/a	n/a
*	TYPE	Type of shadow event: Earth or Lunar	ASCII String	n/a	Earth/ Moon	n/a
	SA_OFFPT	Specifies if SAs should be off-pointed	ASCII String	n/a	Yes|No	n/a
	WINDOW					
	window_start_time	Earliest requested start time	Absolute time	GMT	n/a	n/a
	window_end_time	Latest requested end time	Absolute time	GMT	n/a	n/a
	REPETITION					
	period	Repetition interval	Relative time	n/a	n/a	n/a
	delta	Allowed tolerance in repetition interval	Relative time	n/a	n/a	n/a
	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a
	PRECEDING					
	reqid	Preceding request identification	ASCII string	n/a	1-8 char	n/a
	minimum_lead	Minimum time between requests	Relative time	n/a	n/a	n/a
	maximum_lead	Maximum time between requests	Relative time	n/a	n/a	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	OVERLAP					
	reqid	Overlapping request identification				
	start_lead	Lead time between start of overlapping requests	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping requests	Relative time	n/a	n/a	n/a
	ID	Request Identifier	ASCII string	n/a	n/a	n/a
	pass_through	Pass through parameters	n/a	n/a	n/a	n/a

MIN_DUR — The MIN_DUR parameter specifies the minimum allowed duration for a shadow event. If a shadow event is found with a duration less than this value, the scheduled shadow event support will be artificially extended to this duration by off pointing the solar arrays. This parameter is required.

MAX_DUR — The MAX_DUR parameter specifies the maximum allowed duration for a shadow event. If a shadow event is found with a duration greater than this value, an error message is issued and shadow support is not scheduled. This parameter is required.

ABSMIN_DUR — The ABSMIN_DUR parameter specifies the absolute minimum duration for a shadow event. If a shadow event is found with a duration less than this value, the shadow event will be ignored during scheduling. This is an optional parameter. If a value is not provided for ABSMIN_DUR, the absolute minimum duration will be assumed to be 0 seconds for the specified shadow type and all shadow events of that type will be processed regardless of duration.

TYPE — The TYPE parameter specifies the type of shadow event, Earth or lunar, referenced by the current request. This parameter is required.

SA_OFFPT — This parameter specifies whether the SAs are to be off-pointed when the eclipse duration is greater than or equal to MIN_DUR and less than or equal to MAX_DUR. This is an optional parameter.

WINDOW — The WINDOW parameter is used to provide absolute time

REPETITION — The REPETITION parameter is used to specify the number and interval for repetitive request. The REPETITION parameter has

	three arguments: period specifies the period for the repetition of the request, delta specifies an allowed tolerance in the scheduling of the request, and number specifies the number of times the request is to be repeated. The REPETITION parameter cannot be used if the WINDOW parameter is used to uniquely select a shadow event. This is an optional parameter.
PRECEDING	The PRECEDING parameter specifies a required precedence between requests. The PRECEDING parameter has three arguments: reqid specifies the ID for the request which must precede this request in scheduling order, minimum lea specifies the minimum time delay between the end of the preceding request and the start of this request, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this request. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests, If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.OVERLAP The OVERLAP parameter specifies a request that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specifies the request id for the request that must overlap this request, start lead specifies the maximum delay between the start of this request and the start of the overlapping request, end lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start lead and end lead arguments are omitted, the entire request must bc covered by the referenced overlapping request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
ID	The ID parameter provides a unique identifier for tracing each activity through ground processing and onboard execution. This is an optional parameter. However, ID must be provided if this request is to be referenced in the PRECEDING or OVERLAP parameter of any other request.
pass_through	Pass through parameters may be provided on this statement in the form, parameter name=value. If provided, the parameters will be included in the DOT statement for this request and used by CM as input values for building the commands in this request. The MPS will have no knowledge of the meaning of or utilize any of the pass-through parameters during scheduling.

3.13.3.8 HDR Statement Syntax

HDR, HDR_ID=ASCII, [SCH_ALG=Integer] [,TIME_SPAN=(beg_time,end_time)] [,WEIGHT=(class_id,cl_weight)]

R	Parameter	Description	Type	Units	Range	Resolution
*	HDR_ID	Engineering List Identifier	ASCII string	n/a	1-11	
	SCH_ALG	Scheduling algorithm used (1-20).	Integer	n/a	1-20	n/a
	TIME_SPAN					
	beg_time	Beginning time of oversubscribed ER list. (HOSC GMT)	Absolute time	GMT	n/a	n/a
	end_time	Ending time of oversubscribed ER list. (HOSC GMT)	Absolute time	GMT	n/a	n/a
	WEIGHT					
	class_id	Unique class identifier.	ASCII	n/a	1-8 char	
	cl_weight	Weighting factor for this class.	Integer	n/a	1-10	

The header statement is used to define general information on the schedule such as time span, weighting factors, etc. The allowed HDR parameter definitions are as follows:

HDR_ID — The HDR_ID parameter defines a unique engineering list identifier used to track the engineering list in the ASC and OFLS systems. The last 3 characters are reserved for a two digit revision indicator in the form “_nn”.

SCH_ALG — The SCH_ALG parameter defines the scheduling algorithm to be used. The argument is an Integer value that defines a particular scheduling algorithm (1=maximize time on target; 2=maximize time on target in priority order, etc.)

TIME_SPAN — The TIME_SPAN parameter is used to specify the time span for the schedule. The TIME_SPAN parameter has two arguments: beg_time is the beginning time of the schedule in HOSC GMT, end_time is the end time of the schedule in HOSC GMT.

WEIGHT — The WEIGHT parameter is used to specify the weighting factor for the various classes. The WEIGHT parameter has two arguments: classid is the identifier for the class, cl_weight is the weighting factor for that class. Multiple weight parameters can be specified.

3.13.3.9 BEGIN_COMMENT Statement Syntax

BEGIN_COMMENT[,ID=string]

R	Parameter	Description	Type	Units	Range	Resolution
*	ID	Request Identifier	ASCII string	n/a	1-8 char	n/a

BEGIN_COMMENT

The begin comment statement is used to mark the beginning of a comment. All text that follows, until the END_COMMENT statement, will be considered a comment for that ID. The allowed BEGIN_COMMENT parameter definitions are as follows:

ID — The ID parameter provides a unique identifier that is retained and passed to command management for tracing each observation through ground processing, onboard execution, and post-observation data processing. This is a required parameter for comments that are to be passed to command management.

3.13.3.10 END_COMMENT Statement Syntax

END_COMMENT

END_COMMENT

The END comment statement is used to mark the end of a comment. This statement must be used in conjunction with the BEGIN_COMMENT statement. There are no parameters associated with this statement.

3.14 EPHEMERIS, DEFINITIVE

3.14.1 Header Information

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	OFLS (ISS)
Data Routinely Accessed By:	OFLS
Format Structure:	Binary, Direct Access
Delivery Media	Electronic Transfer

3.14.2 Content

The definitive ephemeris file provides a history of actual AXAF spacecraft position and velocity information based on measured AXAF positions and velocities from DSN ranging data. The definitive ephemeris file covers a time interval from a user-specified time in the past to the last available DSN ranging data. The definitive ephemeris file comprises four data record types. The first two records are header records providing information about the file and the AXAF orbit. The third record format provides the detailed spacecraft position and velocity records and is repeated until all data has been provided. The fourth record indicates the end of the file. The format of each record is provided in the following sections.

3.14.3 Format Description

FILE ORGANIZATION: Direct

FILE SIZE (ESTIMATED): 3,000,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Ephemeris Record 1

Record Format: Fixed

Record Length: 2800 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_tape_id	Input tape ID.	Real		8
odb_ephem_satellite_id	Satellite ID number.	Real		8
odb_ephem_utc_flag	UTC flag.	Real		8
odb_ephem_start_date	Date of start of ephemeris.	Real		8
odb_ephem_start_day_count	Day count of year for start of ephemeris.	Real		8
odb_ephem_start_sec_count	Seconds of day for start of ephemeris.	Real		8
odb_ephem_stop_date	Date of end of ephemeris.	Real		8
odb_ephem_stop_day_count	Day count of year for end of ephemeris.	Real		8
odb_ephem_stop_sec_count	Seconds of day for end of ephemeris.	Real		8
odb_ephem_step_size	Time interval between ephemeris.	Real		8
odb_ephem_rec1_spare1	Field not used.	Real		136
epm_ref_date	Reference date.	Real		8
odb_ephem_coord_type	Coordinate system type indicator.	Integer		4
odb_ephem_rec1_spare2	Field not used.	Real		132
epm_epoc_time	Epoch time (in DUT).	Real		8
odb_ephem_epoch_year	Year of element epoch.	Real		8
odb_ephem_epoch_month	Month of element epoch.	Real		8
odb_ephem_epoch_day	Day of element epoch.	Real		8
odb_ephem_epoch_hour	Hour of elements epoch.	Real		8
odb_ephem_epoch_min	Minute of element epoch.	Real		8

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_epoch_ milisec	Milliseconds of element epoch.	Real		8
odb_ephem_smajor_axis	Semi-major axis at t0 (km).	Real		8
odb_ephem_eccent	Eccentricity at t0.	Real		8
odb_ephem_inclin	Inclination at t0 (rad).	Real		8
odb_ephem_perigee	Argument of perigee at t0 (rad).	Real		8
odb_ephemraan	Right ascension of ascending node at t0 (rad).	Real		8
odb_ephemmena_anom	Mean anomaly at t0 (rad).	Real		8
odb_ephemtrue_anom	True anomaly at t0 (rad).	Real		8
odb_ephemsum_aprgta	Sum of argument of perigee and true anomaly.	Real		8
odb_ephemrec1_spare6	Field not used.			16
odb_ephemperiod	Period at epoch.	Real		8
odb_ephemrec1_spare7	Field not used.			16
odb_ephemmean_motn	Mean motion.	Real		8
odb_ephemrec1_spare8	Field not used.			8
odb_ephemrate_ascnd	Rate of change of RA off ascending node.	Real		8
odb_ephem_pos_vector	Position vector (x,y,z) at t0.	Real	3	24
odb_ephem_vel_vector	Velocity vector (dx,dy,dz) at t0.	Real	3	24
odb_ephem_rec1_spare3	Field not used.	Real		456
odb_ephemsolar_pos	Solar position vector.	Real	3	24
odb_ephemrec1_spare4	Field not used.			520
odb_ephem_grhour_angl e	Greenwich hour angle at t_0 (rad).	Real		8
odb_ephemrec1_spare5	Field not used.			1200

RECORD LAYOUT:

Record Identifier: Ephemeris Record 2

Record Format: Fixed

Record Length: 2800 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_rec2_spare1	Dummy header record.	Integer		2800

RECORD LAYOUT:

Record Identifier: Ephemeris Record 3

Record Format: Fixed

Record Length: 2800 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_date_first_point	Date of first ephemeris point.	Real		8
odb_ephem_days_in_year_first_point	Day count of year for first ephemeris point.	Integer		8
odb_ephem_sec_in_day_first_point	Seconds of day for first ephemeris point.	Integer		8
odb_ephem_step_time	Time interval between data points (seconds).	Relative time		8
odb_ephem_first_pos_vector	First position vector (x,y,z) (10^4 kilometers).	Real	3	24
odb_ephem_first_vel_vector	First velocity vector (dx,dy,dz) (10^4 kilometers/(0.01 day)). **Note**: Within the binary ephemeris file that OFLS creates, position vectors are stored in units of 10^4 km and velocity vectors as 10^4 km/(0.01 day). Thus, to obtain position in km, multiply the position values from the file by 10^4. To obtain velocity in km/sec, divide the velocity values from the file by 0.0864. The origin of such units date back to the 60's when various computational trade-offs were made among speed, accuracy, dynamic range, etc.	Real	3	24

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_pos_vel_data_2_50	Position and velocity vector sets for data points 2 through 50 (if there are less than 50 ephemeris data points, the first data point following the last valid data point consists of a set of 6 values equal to $0.9999999999999999 \times 10^{16}$).	Real	49	2352
odb_ephem_rec3_spare1	Field not used.	Integer		368

RECORD LAYOUT:

Record Identifier: Ephemeris Record 4

Record Format: Fixed

Record Length: 2800 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_ephem_sentinels	Ten sentinels-$0.9999999999999999 \times 10^{16}$).	Real	10	80
odb_ephem_rec4_spare1	Field not used.	Integer		2720

3.15 Ephemeris, Predictive

3.15.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (ISS)

Data Routinely Accessed By: OFLS

Format Structure: Binary, Direct Access

Delivery Media Electronic Transfer

3.15.2 CONTENT

The predictive ephemeris file provides estimated AXAF spacecraft position and velocity information based on the last measured AXAF position and velocity from DSN ranging data. The predictive ephemeris file covers a time interval from the last available DSN ranging data to a user-specified time in the future. The predictive ephemeris file comprises four data record types. The first two records are header records providing information about the

file and the AXAF orbit. The third record format provides the detailed spacecraft position and velocity records and is repeated until all data has been provided. The fourth record indicates the end of the file.

3.15.3 FORMAT DESCRIPTION

The format of each record in the predictive ephemeris file is identical to the format for the definitive ephemeris file defined in section 3.14.

3.16 MCILWAIN PARAMETERS

3.16.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	OFLS (ISS)
Data Routinely Accessed By:	ASC
Format Structure:	ASCII, Sequential
Delivery Media	Electronic Transfer

3.16.2 CONTENT

The McIlwain parameters file provides a listing of McIlwain parameters and the times at which the parameters were calculated. McIlwain parameters are recalculated and provided for each definitive ephemeris file.

3.16.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 16,000 (Bytes)

RECORD LAYOUT:

Record Identifier: McIlwain Parameters Record

Record Format: Fixed

Record Length: 16 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_mcilwain_gmt	Time at which McIlwain is calculated (extended HOSC GMT).	ASCII		21
odb_mcilwain_parameter	McIlwain parameter.	Real		8

3.17 MEMORY IMAGE, AC

3.17.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.17.2 CONTENT

The AC Memory Image data element contains memory image of the Aspect Camera Assembly. There are 2 ACA's (A side and B side). The memory maps are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.17.3 FORMAT DESCRIPTION

Refer to SMF to OCC ICD for format definitions.

3.18 MEMORY IMAGE, CPE

3.18.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.18.2 CONTENT

The CPE Memory Image data element contains memory image of the Control Processor Electronics (CPE). There are 2 CPE's (A and B side). The memory images are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.18.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.19 MEMORY IMAGE, CTU EEPROM

3.19.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.19.2 CONTENT

The CTU EEPROM Memory Image data element contains the Electrically Erasable Programmable Read Only Memory (EEPROM) image of the Command and Telemetry Unit (CTU). There are 2 CTU's (A and B side). The memory images are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.19.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.20 MEMORY IMAGE, I-EPHIN

3.20.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.20.2 CONTENT

The I-EPHIN Memory Image data element contains the memory images of the Integrated-Electron Proton Helium Instrument (I-EPHIN). The I-EPHIN consists of the EPHIN and EIO. The memory maps are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.20.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.21 MEMORY IMAGE, IU EEPROM

3.21.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.21.2 CONTENT

The IU EEPROM Memory Image data element contains the EEPROM memory images of the Interface Unit (IU). There are 2 IU's (A and B side), each IU contains 2 program images. The memory maps are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.21.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.22 MEMORY IMAGE, OBC

3.22.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.22.2 CONTENT

The OBC Memory Image data element contains memory image of the On-Board Computer (OBC). There are 2 OBC's (A and B side). The memory maps are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.22.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.23 MEMORY IMAGE, SIM

3.23.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	ONLS (Mission Comp)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.23.2 CONTENT

The SIM Memory Image data element contains memory image of the Science Instrument Module(SIM). There are 2 SIM hardware (A and B side) each contains a RAM and a ROM. The memory maps are retrieved from a telemetry dump process by the ONLS. The image can be used for the Dump and Compare process.

3.23.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.24 OBSERVATION REQUESTS

3.24.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	ASC
Data Routinely Accessed By:	OFLS (MPS)
Format Structure:	ASCII, Sequential
Delivery Media:	Electronic Transfer

3.24.2 CONTENT

The ODB/ODE OR list contains individual requests for each desired science observation. Each statement can span multiple lines and comprises a keyword indicating the statement type, parameters associated with that keyword, and values for the parameters.

Requirements and constraints for the roll angle, target, target ID, science instrument, start/stop times, segment duration, cumulative duration, requested duration, gratings, and target offsets, can be included in the OR.

The OR List can have any name, but must end with the extension .OR.

3.24.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 50,000 (Bytes)

RECORD LAYOUT:

Record Identifier: OR Request Record

Record Format: Fixed

Record Length: 80 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_orreq_line	Defines one line of OR request input.	ASCII		80

3.24.3.1 OR Statement Definitions

The syntax for each OR statement is defined by a sample statement format listing all required and optional parameters, a parameter table describing each parameter, and descriptive text paragraphs for each parameter.

SAMPLE STATEMENT FORMAT

Some or all of the parameters are presented using the correct statement syntax. The following conventions apply

|	A vertical bar is used to indicate selection between one or more mutually exclusive options
[ ]	Square brackets are used to enclose optional parameters.

PARAMETER TABLE

Each of the parameters are presented with the following information:

Column 1 R Required parameters and arguments. An asterisk in this column indicates parameters that are required with each request or arguments that are required with a parameter.

Column 2 Parameter Capital letters indicate parameter names. Lower case letters indicate arguments for the parameter. Brackets are used to enclose mutually exclusive parameters and subparameters. Parameter names are always provided on the statement. Values are substituted for argument names in the statement.

Column 3 Description Short description of each parameter or argument.

Column 4 Type Indicates type of value to be provided with each parameter or argument. Valid types are ASCII string, mnemonic, real, integer, absolute time or integer time.

Column 5 Units Indicates units in which parameter or argument are to be provided.

Column 6 Range Indicates allowed range of values for each parameter or argument. Vertical bars indicate mutually exclusive choices between allowed values. Underscores indicate default value (if any).

Column 7 Resolution Indicates required precision which must be allowed for each parameter or argument.

Text statement descriptions list each parameter, the allowed values for each parameter or argument, restrictions on providing the parameter or argument, and MPS relations to various combinations of parameters and values.

3.24.3.2 OR Request Syntax Generic Information

The following pages provide the format and contents of ORs. An OR list consists of a special OR list header and separate statements for each OR. OR list statements include the OR list header, OR statements, and delimiters used in the construction of compound statements such as BEGIN_COMMENT and END_COMMENT. The following sections define in detail each statement, including parameter names, allowable parameter values, and the semantics of each statement. The statements are defined three ways: sample statement format, parameter table, and text descriptions of each parameter.

STATEMENT

An OR list statement is defined to be any OR list construct which must begin on a new OR list record. The statements allowed in an OR list are OBS, HDR, BEGIN_COMMENT, and END_COMMENT. Statements consist of ASCII characters and may be of varying lengths contained in 80-character records. Generally, the statement will include a keyword, required and optional parameters, parameter values (mnemonics, strings, values), and selected delimiters. If a statement exceeds 80 characters, continuation records can be used. Continuations must break the statement at a delimiter (comma; fields between commas cannot be broken across record boundaries). Lines that do not end in a delimiter indicate the end of the statement. Blanks have no meaning for the OR list syntax. They can be placed anywhere in the statement and are ignored during input translation. BEGIN_COMMENT and END_COMMENT are compound statements which allow special processing of comments. Comments are retained during OFLS process, are accessed graphically through the mission schedule and detailed operations timeline display tools, and are returned in the command timeline report. If it is desired to have comments in the command timeline report, the comments must be defined within the BEGIN_COMMENT and END_COMMENT statements, with a link to a specific OR in the OR list.

PARAMETER

Each parameter has a single value, specified in the syntax PARAMETER_NAME=argument, or a set of values indicated by a list of parameters, specified in the syntax, PARAMETER_NAME= (argument,argument,...).

The following conventions apply to all of the presentation formats:

absolute time	An ASCII string specifying GMT or an orbit event time. GMT times are specified in HOSC standard format as yyyy:ddd:hh:mm:ss.sss . Milliseconds may be omitted and a

value of 0 will be assumed. Or, an ASCII string specifying an orbit relative time in the format (ORB, orbit_num, orbit_event, +/- relative time).

y - Represents four digits for a year. If y is not defined, then the time is assumed to be relative.

d - Represents three digits for a day, and cannot exceed 366 days and is measured from Greenwich midnight, December 31, preceding the year specified.

h - Represents two digits for hours. Hours are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time.

m - Represents two digits for minutes. Minutes are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time.

s - Represents two digits for seconds and three digits for milliseconds after the decimal. Seconds are measured from Greenwich midnight when used as absolute time or orbit relative time and measured from the orbital event when used as an orbit relative time.

Orbit event times are specified in the format (ORB,orbit_number,orbit_event, +/-relative time). The processing resolves the statement time by referencing the associated time for the specified event in the orbit event file and then applies an optional positive or negative relative time.

relative time	An ASCII string specifying a delta GMT in the format ddd:hh:mm:ss.sss. The day field and/or the millisecond field may be omitted and a value of 0 will be assumed for that field. The day field can have a value from 0 to 999. If the day field is omitted, the hour field can have a value from 0 to 99. If the day field is included, the hour field can only have values from 0 to 23. The minute and seconds fields can have a value from 0 to 59. The milliseconds field can have a value from 0 to 999.
mnemonic	Certain statements take a mnemonic as an argument. The term mnemonic, as used, is a string of alphanumeric and selected special characters with the first character being a letter or the numerals 0 through 9. Letters are required to be uppercase alphabetic characters (A, B, ...Z). The special character _ (underline) is allowed in positions 2 through 8. The mnemonic may provide an index into a look up table for a set of related information. Examples of mnemonics are the SI mode mnemonic on an ACIS OBS used as an index into the ACIS table parameters file in the ODB and the observation id used as an index into the observation request list for required preceding observations.
string	A string is a special form of mnemonic which may start with a special character or number.

text string	Text string is a general purpose quoted line of text (used in several statements) which allows comments or descriptive text such as an argument. The allowable characters in a text string include the ASCII characters decimal 32 to 126 excluding the Single Quote (') and the ampersand (&). The ampersand is reserved for substitution parameter use in command set definitions.
unrestricted string	The x-ray target name on OBS and CAL statements uses a special string that allows use of an extended set of ASCII characters. The unrestricted string is a general purpose text string enclosed in { } (curly brackets). Any legal ASCI character can be used in the string, including characters that are normally restricted, such as the & (ampersand), the . (period), math symbols (+ and -), and the ' (single quote). All text between the { (opening bracket) and the next } (closing bracket) will be accepted, except for another opening bracket. A closing bracket that is not preceded by an opening bracket will be flagged as an error. This string type is only valid in OR and ER lists and cannot be used in command sequence definition files or relative time sequence definition files.
integer	The use of unsigned integers is less restrictive than the hex, octal, and binary notations. The maximum value for the integer is 2**32-1. Integers may not be substituted for decimal floating point numbers or vice versa.
decimal	Decimal floating point numbers may have a field width of up to the 80 character statement line limit. When decimal is specified as an argument type, the decimal point is required.
signed integer	The range of values for a signed integer is -2**32+1 to 2**32-1. Signed integers may not be substituted for decimal floating point numbers or vice versa.
hex, octal, binary	Hex, octal, and binary number arguments are allowed to specify up to 32 bits of information with no sign bit recognition. These forms may be used as input to command statements, tables, and indirect commands when the allowable values are unsigned integers. The only other place where these are allowed are as arguments used in substitution into command sets using the same above statements.

3.24.3.3 OBS Statement Syntax

OBS, ID=integer, TARGET=(ra, dec, name)|MANEUVER=(v1, v2, v3, angle, ref)
|SS_OBJECT=string[, SL_RATE=real], DURATION=(nominal[,min_duration][,
max_duration]), PRIORITY=integer, SI=ACIS-I|ACIS-S|HRC-I
|HRC-S[,STAR=(ra,dec,mag,type,id] , MIN_GUIDE, MIN_ACQ [,FID=(fidid, mag),
[,GRATING=HETG|LETG |NONE] [,SI_MODE=mnemonic] [,BIAS=mnemonic]

[,ACA_MODE=mnemonic] [,CLASS=string] [,ROLL=(roll_angle,roll_tolerance)] [,TARGET_OFFSET=(y_offset, z_offset)] [,SIM_OFFSET=(trans_offset, focus_offset)] [,DITHER=(status, y_amp, y_freq, y_phase , z_amp, z_freq, z_phase,)] [,WINDOW=(window_start_time,window_end_time)],[,PHASE=(period, epoch, start_range, start_margin, end_range, end_margin)] [,REPETITION=(period, delta, number_reps)] [,PRECEDING=(reqid, minimum_lead, maximum_lead)] [,OVERLAP=(reqid, start_lead, end_lead)] [,SEGMENT=(max_number, min_duration, max_separation)] [,MOON=(status, avoid_angle)] [,SUN=(status, avoid_angle)] [,FSS=(status, fss_fov_angle)] [,EARTH=(status, avoid_angle)] [,PLANET=(id, status, avoid_angle)] [,OBJECT=(id, status, avoid_angle)], [,E_RADIATION=(status, energy, flux)][, P_RADIATION=(status,energy, flux)] [,ECLIPSE=(status,DAY|NIGHT)] [,OVERLAP=(reqid, start_lead, end_lead)] [,ALTITUDE=(status, min_altitude, max_altitude)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**ID**	Request Identifier	Integer	n/a	0≤x<5500 0	n/a
*	{ **TARGET**					
*	ra	Target Position Right Ascension	Real	degrees	0.0 - 360.00-360	n/a
*	dec	Target Position Declination	Real	degrees	-90.0 - +90.0	n/a
	name	Target name	Unrestricted string	n/a	1-20 char	n/a
*	**MANEUVER**					
*	v1	Maneuver Unit Eigenvector Component 1	Real	n/a	0 - 1	n/a
*	v2	Maneuver Unit Eigenvector Component 2	Real	n/a	0 - 1	n/a
*	v3	Maneuver Unit Eigenvector Component 3	Real	n/a	0 - 1	n/a
*	angle	Maneuver Angle	Real	degrees	0.0 - 360.0	n/a
	ref	Reference frame the eigenvector is defined in	ASCII	n/a	GCI| SC	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	**SS_OBJECT**	Name of solar system object to view (defined in the solar, lunar, planetary ephemeris)	ASCII	n/a	*EARTH*| *MOON*| *MERCUR Y*| *VENUS*| *MARS*| *JUPITER*| *SATURN*| *URANUS*| *NEPTUN E*| *PLUTO*| *SUN*	n/a
	MT_OBJECT}					
*		Name of solar system object to view (defined in the ODE-Object_character istics)	ASCII	n/a	*1-20 char*	n/a
	SL_RATE	Defines the rate at which the slew to the target is to be performed	real	degrees/s econd	n/a	n/a
*	**DURATION**					
*	nominal	Observation Duration	Real	seconds	n/a	
	min_duration	Minimum Observation Duration	Real	seconds	n/a	
	max_duration	Maximum Observation Duration	Real	seconds	n/a	
*	**PRIORITY**	Observation Priority (Highest to Lowest)	Integer	n/a	1 - 10	n/a
*	**SI**	Observation Science Instrument	ASCII string	n/a	ACIS-I| ACIS-S| HRC-I| HRC-S	n/a
	STAR					
*	ra	Star right ascension	Real	degrees	0.0 - 360.0	0.0001
*	dec	Star declination	Real	degrees	-90.0 - +90.0	0.0001
*	mag	Star instrumental magnitude	Real	mag	-10.0 - +20.0	0.01

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	type	Star type: guide, acquisition, monitor or both guide and acquisition	ASCII	n/a	GST| ACQ| BOTH|MON	n/a
	id	Star catalog id	Integer	n/a	-1e8 to +2e9	n/a
	MIN_GUIDE	Minimum number of guide stars to be selected	Integer	n/a	0 - 8	n/a
	MIN_ACQ	Minimum number of acquisition stars to be selected	Integer	n/a	0 - 8	n/a
	FID					
*	fidid	FID light identifier	Integer	n/a	1-14	
	mag	FID light instrumental magnitude	Real	mag	-10.0 - +20.0	0.01
	GRATING	Observation Grating	ASCII string	n/a	LETG| HETG |NONE	n/a
	SI_MODE	Science Observation Mode	Mnemonic	n/a	1-10 char	n/a
	BIAS	ACIS Bias Option	Mnemonic	n/a	OPT|REQ| NONE	n/a
	ACA_MODE	ACA Request Mode	Mnemonic	n/a	1-8 char |NONE	n/a
	CLASS	Observation Class	ASCII string	n/a	TBD	n/a
	ROLL					
*	roll_angle	Target Roll	Real	degrees	0 - 360.0	n/a
	roll_tolerance	Target Roll Tolerance	Real	degrees	n/a	n/a
	TARGET_OFFSET					
	y_offset	Target position offset in the y-direction (rotation around the z-axis)	Real	degrees	-2.0 - +2.0	n/a
	z_offset	Target position offset in the z-direction (rotation around the y-axis)	Real	degrees	-2.0 to +2.0	n/a
	SIM_OFFSET					

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	trans_offset	Offset from nominal of translation position. Range limits are currently set based on the SIM mechanical hard stops and the halfway point between adjacent SIs.	Integer	SIM translation steps	ACIS-I: -8976 to +10927 ACIS-S: -75766 to +8976 HRC-I: -24401 to +50360 HRC-S: -4989 to +24401	SIM Motor Step
	focus_offset	Offset from nominal of focus position. Range limits are currently set based on the SIM mechanical hard stops the overshoot for SIM focus hysteresis, and the halfway point between adjacent SIs.	Integer	SIM focus motor steps	ACIS-I: -11463 to +6612 ACIS-S: -11463 to +6612 HRC-I: -11463 to +6612 HRC-S: -11463 to +6612	SIM Motor Step
	DITHER					
	status	Status of onboard dither processing	Mnemonic	n/a	ON|OFF	n/a
	y_amp	Spacecraft dither amplitude in the y-axis direction of the spacecraft body frame	Real	deg	0.0000 ~ +5.555E-3	0.0001
	y_freq	Spacecraft dither frequency in the y-axis direction of the spacecraft body frame	Real	deg/sec	0.0001 ~ +1.5758	0.0001
	y_phase	Spacecraft dither phase in the y-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +360.0000	0.0001

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	z_amp	Spacecraft dither amplitude in the z-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +5.555E-3	0.0001
	z_freq	Spacecraft dither frequency in the z-axis direction of the spacecraft body frame	Real	deg/sec	0.0001 ~ +1.5758	0.0001
	z_phase	Spacecraft dither phase in the z-axis direction of the spacecraft body frame	Real	deg	0.0001 ~ +360.0000	0.0001
	WINDOW					
	window start time	Earliest Requested Observation Start Time	absolute time	GMT	n/a	n/a
	window end time	Latest Requested Observation End Time	absolute time	GMT	n/a	n/a
	PHASE					
*	period	Object ephemeris Period	relative time	GMT	n/a	n/a
*	epoch	Absolute Time for Phase 0	absolute time	GMT	n/a	n/a
	start_range	Start range of phase to be observed	Real	phase	0 - 1	n/a
	start_margin	Allowed range in phase interval starting point	Real	phase	0 - 1	n/a
	end_range	End range of phase to be observed	Real	phase	0 - 1	n/a
	end_margin	Allowed range in phase interval ending point	Real	phase	0 - 1	n/a
	REPETITION					
*	period	Repetition Interval	relative time	GMT	n/a	n/a
	delta	Allowed Tolerance in Repetition Interval	relative time	GMT	n/a	n/a
*	number	Number of repetitions	Integer	n/a	1 to +2e9	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	PRECEDING					
*	reqid	Preceding Request Identification	ASCII string	n/a	1-8 char (TBR)	n/a
	minimum_lead	Minimum Time Between Observations	relative time	GMT	n/a	
	maximum_lead	Maximum Time Between Observations	relative time	GMT	n/a	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a
	start_lead	Lead time between start of overlapping requests.	relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	relative time	n/a	n/a	n/a
	SEGMENT					
	max_number	Maximum Number of Segments	Integer	n/a	>0	n/a
	min_duration	Minimum Duration of Any Segment	Real	Seconds	n/a	n/a
	max_separation	Maximum Separation Between Segments	Real	Seconds	n/a	n/a
	MOON					
*	status	Status of Moon Occultation Check	ASCII string	n/a	ON|OFF	n/a
*	avoid_angle	Moon Occultation Check Angle	Real	degrees	0.0 - 360.0	TBD
	SUN					
*	status	Status of Sun Occultation Check	ASCII string	n/a	ON|OFF	n/a
*	avoid_angle	Sun Occultation Check Angle	Real	degrees	0.0 - 360.0	TBD
	FSS					
	status	Status of FSS FOV check	ASCII string	n/a	ON|OFF	n/a
	fss_fov_ angle	FSS FOV check angle	Real	degrees	0.0 - 360.0	n/a
	EARTH					

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	status	Status of Earth Occultation Check	ASCII string	n/a	ON|OFF	n/a
*	avoid_angle	Earth Occultation Check Angle	Real	degrees	0.0 - 360.0	TBD
	PLANET					
*	id	Planet Identification	Mnemonic	n/a	1-8 char (TBR)	n/a
*	status	Status of Planet Occultation Check	ASCII string	n/a	ON|OFF	n/a
*	avoid_angle	Planet Occultation Check Angle	Real	degrees	0.0 - 360.0	TBD
	OBJECT					
*	id	X-Ray Object Identification	Mnemonic	n/a	1-8 char (TBR)	n/a
*	status	Status of Object Occultation Check	ASCII string	n/a	ON|OFF	n/a
*	avoid_angle	Object Check Angle	Real	degrees	0.0 - 360.0	TBD
	E_RADIATION					
*	status	Status of Electron Radiation Check	ASCII string	n/a	ON|OFF	n/a
*	energy	Electron Energy Level	Real	MeV	n/a	n/a
*	flux	Electron Flux Level	Real	con^2sec	n/a	n/a
	P_RADIATION					
*	status	Status of Proton Radiation Check	ASCII string	n/a	ON|OFF	n/a
*	energy	Proton Energy Level	Real	MeV	n/a	n/a
*	flux	Proton Flux Level	Real	con^2sec	n/a	n/a
	ECLIPSE					
*	status	Status of Eclipse (Day/Night) Check	ASCII string	n/a	ON|OFF	n/a
*	day/night	Indicates Desired Eclipse Condition for Scheduling Observation	ASCII string	n/a	DAY| NIGHT	n/a
	OVERLAP					
	reqid	Overlapping request identification.	ASCII string	n/a	1-5 char	n/a

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
	start_lead	Lead time between start of overlapping requests.	Relative time	n/a	n/a	n/a
	end_lead	Lead time between end of overlapping request	Relative time	n/a	n/a	n/a
	ALTITUDE					
	status	Status of altitude check	mnemonic	n/a	ON|OFF	n/a
*	min_altitude	Minimum altitude at which request can be scheduled	Real	km	n/a	n/a
*	max_altitude	Maximum altitude at which request can be scheduled	Real	km	n/a	n/a

OBS

The observation statement is used to request specific science observations and specify the mode of operation and special observation constraints associated with the observation. Observations can be specified in terms of a specific target location or object on the sky or in terms of maneuver across a portion of the sky. The allowed OBS parameter definitions are as follows:

ID — The ID parameter provides a unique identifier that is retained and passed to command management for tracing each observation through ground processing, onboard execution, and post-observation data processing. This is a required parameter.

TARGET — The TARGET parameter is used to specify the target location for an observation. The TARGET parameter has three arguments: ra is the right ascension in J2000 coordinates of the requested target, dec is the declination in J2000 coordinates of the requested target. name is the object name of the requested target. The sub-parameters ra and dec are required; the sub-parameter name is optional. Either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each observation.

Multiple TARGET parameters may be specified. However, TARGET parameters may not be mixed with MANEUVER, SS_OBJECT and MT_OBJECT parameters on a single observation statement. If multiple TARGET parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each TARGET parameter may be scheduled as a maneuver or a nudge based on the

maneuver angle, sun to solar array normal angle, and the number of guide stars retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

MANEUVER

The MANEUVER parameter is used to specify a maneuver across an extended region of the sky. The MANEUVER parameter has five arguments: v1, v2, and v3 are the components of the unit vector specifying the eigenaxis of the maneuver; angle is the maneuver angle about the eigenaxis, ref identifies whether the eigenaxis is defined in the GCI or the spacecraft reference frame.

Multiple MANEUVER parameters may be specified. However, MANEUVER parameters may not be mixed with TARGET, SS_OBJECT and MT_OBJECT parameters on a single observation statement. If multiple MANEUVER parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each TARGET parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide stars retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

SS_OBJECT

The SS_OBJECT parameter is used to specify a solar system object (such as the Moon, Jupiter, etc.) as the viewing target. The SS_OBJECT must be defined in the SLP file. The object position computed from the SLP data is centered in the sensor FOV at the mid-point of the observation. Either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each observation.

Multiple SS_OBJECT parameters may be specified. However, SS_OBJECT parameters may not be mixed with MANEUVER, TARGET and MT_OBJECT parameters on a single observation statement. If multiple SS_OBJECT parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each TARGET parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide stars retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).

SL_RATE	The RATE parameter specifies how fast the slew to the requested target is performed. The SL_RATE parameter can be used with either the TARGET, MANEUVER, or SS_OBJECT parameter. The value for the SL_RATE parameter is dependent on the value of the SLEW_HW parameter, if provided, and must be greater than 0 and less than or equal to the maximum slew rate specified in the AXAF Constraints. This is an optional parameter. If no value is provided, the rate is assumed to be the maximum allowed slew rate.
DURATION	The DURATION parameter specifies the requested duration for the observation. The DURATION parameter has three arguments: nominal is the desired duration, min_duration is the minimum duration allowed, and max_duration is the maximum duration allowed. The observation may be extended beyond the nominal duration in order to utilize spacecraft idle time during the scheduling process, however it may not extend past the maximum duration. The min_duration and max_duration subparameters are optional. If omitted, the request will be scheduled at the nominal duration with no adjustment. The DURATION parameter and the nominal subparameter are required.
PRIORITY	The PRIORITY parameter specifies the priority of this observation for scheduling. The priority is an Integer number from 1 to 10 (1 is the highest scheduling priority) and will be used by all scheduling algorithms for which the goal includes consideration of observation priority. This is a required parameter.
SI	The SI parameter specifies the requested science instrument to be used for the observation. Allowed values are ACIS-I, ACIS-S, HRC-I or HRC-S. This is a required parameter.
STAR	The STAR parameter specifies an guide star or acquisition star. The star parameter has five arguments: ra is the right ascension in J2000 coordinates, dec is the declination in J2000 coordinates, mag is the instrumental magnitude of the star, type specifies whether the star is a guide star, an acquisition star, a monitor window, or is used for both acquisition and guide star determination, and id specifies the star catalog identifier for the star (if any). Multiple STAR parameters may be specified. The total number of STAR plus FID parameters may not exceed 13. This is an optional parameter.
FID	The FID parameter specifies a fid light. The FID parameter has two arguments: fidid is the fiducial light identifier and mag is the instrumental magnitude of the fid lights. Multiple FID parameters may be specified. The total number of STAR plus FID parameters may not exceed 13. This is an optional parameter.

MIN_GUIDE	The MIN_GUIDE parameter specifies the minimum number of guide stars to be selected for the current request. If the value for MIN_GUIDE is greater than 5, then the number of FID lights selected will be (8 - MIN_GUIDE). If guide stars are specified by the STAR parameter and MIN_GUIDE is specified, then the number of guide stars selected will be at least (MIN_GUIDE - number of guide stars specified). This parameter is optional. If MIN_GUIDE is not specified, the software will use the default value in the ODE Characteristics.
MIN_ACQ	The MIN_ACQ parameter specifies the minimum number of acquisition stars to be selected for the current request. If acquisition stars are specified by the STAR parameter and MIN_ACQ is specified, then the number of guide stars selected will be at least (MIN_ACQ - number of acquisition stars specified). This parameter is optional. If MIN_ACQ is not specified, the software will use the default value in the ODE Characteristics.
MT_OBJECT	The MT_OBJECT parameter is used to specify a solar system object (such as an asteroid or a comet) as the viewing target. The MT_OBJECT must be defined in the ODE_Object_Characteristics record of the Characteristics file. The object position computed from the orbital elements in the ODE is centered in the sensor FOV at the mid-point of the observation. Either TARGET or MANEUVER or SS_OBJECT or MT_OBJECT are required for each observation. Multiple MT_OBJECT parameters may be specified. However, MT_OBJECT parameters may not be mixed with MANEUVER, TARGET, and SS_OBJECT parameters on a single observation statement. If multiple MT_OBJECT parameters are specified, the target specified is assumed to be in the FOV for the duration period specified in the DURATION parameter. The change in spacecraft attitude between each TARGET parameter may be scheduled as a maneuver or a nudge based on the maneuver angle, sun to solar array normal angle, and the number of guide stars retained. If a spacecraft maneuver is required, the target is assumed to move at a rate such that the target will still be in the FOV at the end of the maneuver and star acquisition (~13 to 16 minutes).
GRATING	The GRATING parameter specifies the transmission grating to be used for the observation. Allowed values are HETG, LETG, or NONE. This is an optional parameter.
SI_MODE	The SI_MODE parameter specifies the operational mode for ACIS or HRC observations depending on the SI parameter. This parameter is a mnemonic used for a look-up into the table of modes to obtain the correct table parameters for the observation.

	A default operational mode is provided using the mnemonic DEFAULT. The schedule generation software will maintain history on the operational mode and provide the table parameters for uplink to the spacecraft if the operational mode changes between consecutively scheduled observations. This is an optional parameter.
BIAS	The BIAS parameter specifies the bias measurement options for ACIS observations. Allowed values are OPT, REQ, or NONE. If OPT is specified, then the MPS determines if a bias is needed based on the scheduling rules. If REQ is specified, then a bias measurement will always be scheduled. If NONE is specified, then no bias measurement will be scheduled. This is an optional parameter. The BIAS parameter can only be specified if the value of the parameter SI is ACIS-I or ACIS-S.
ACA_MODE	The ACA_MODE parameter specifies the operational mode for ACA observations. This parameter is a mnemonic used for a look-up into the table of modes to obtain the correct table parameters for the observation. If the ACA_Mode parameter is not provided on the statement, a default mode of NONE is assumed and the observation request is performed without ACA support.
CLASS	The CLASS parameter specifies the class of observations to which this observations belongs. The CLASS parameter is used by certain scheduling algorithms to provide balanced coverage between the different types of observations. The allowed values are TBD. This is not a required parameter.
ROLL	The ROLL parameter specifies a specific spacecraft roll for the observation in order to align the target correctly within the science aperture. The ROLL parameter has two arguments: roll_angle is the position angle of the AXAF-Z axis projected onto the sky at the nominal boresight pointing of AXAF measured from North through East and roll_tolerance is the allowed tolerance of the roll angle. ROLL is specified in decimal degrees in celestial coordinates. Because of roll constraints to maintain proper Sun pointing, the roll parameter will constrain the time window which within the observation can be scheduled. This is an optional parameter; a ROLL may only be supplied if the TARGET parameter was supplied.
TARGET_OFFSET	The TARGET_OFFSET parameter specifies an offset between the spacecraft boresight and the target position specified by the TARGET parameter. The TARGET_OFFSET parameter has two arguments: y_offset is the offset along the y-axis (rotation around the z-axis in the SI reference frame), z_offset is the offset along the z-axis (rotation around the y-axis in the SI reference frame).

	A positive offset in either the y or z direction causes the spacecraft boresight to be offset negatively in the y or z direction, respectively. This is an optional parameter; a TARGET_OFFSET may only be supplied if the TARGET parameter was supplied.
SIM_OFFSET	The SIM_OFFSET parameter specifies an offset from the nominal SIM position. The SIM_OFFSET parameter contains two arguments: trans_offset is the offset from nominal of translation position and focus_offset is the offset from nominal of focus position. This parameter is optional.
DITHER	The DITHER parameter specifies the dither mode for the observation. The DITHER parameter has 7 arguments. Status indicates whether spacecraft dither is on or off for the observation. The six parameters: y_amp, y_freq, y_phase, z_amp, z_freq, z_phase, provide values for the spacecraft y-axis amplitude, frequency, and phase and the spacecraft z-axis amplitude, frequency, and phase, respectively. The schedule generation software will maintain history on the current commanded dither mode and values and provide the dither command parameters for uplink to the spacecraft if either the dither mode or the dither parameters change between consecutively schedule observations. This is an optional parameter. If the parameter is omitted, the observation will default to dither on using the nominal dither amplitude, frequency, and phase from the AXAF characteristics. If the status is specified as on and any or all of the six dither parameters are omitted, the values will default to the nominal values specified in the AXAF characteristics. If the status is specified as off, any additional dither parameters will be ignored.
WINDOW	The WINDOW parameter is used to provide absolute time constraints on the scheduling of the observation. The WINDOW parameter has two arguments: window start time and window end time. The observation must be scheduled to start on or after the window start time and complete on or before the window end time. The WINDOW parameter can be used to completely time constrain the scheduling of the observation by providing a window which is equal to the minimum duration of the observation as specified in the DURATION parameter. The WINDOW parameter can also be used to partially time constrain the scheduling of the observation. Specifying window start and end times that are less than the observation duration, indicates that the observation must cover the specified interval but places no restrictions on when the interval must occur during the observation. Omitting the window end argument indicates that the observation can be scheduled at any time after the specified time. Omitting the window start

	argument indicates that the observation must complete any time prior to the specified time. This is an optional parameter.
PHASE	The PHASE parameter specifies time constraints on the scheduling of the observation based on the ephemeris of the observed object. The PHASE parameter has six arguments: period specifies the phase period of the object, epoch specifies the absolute time for phase 0, start_range specifies the start of the phase, end_range specifies the end of the phase, start_margin specifies an allowed tolerance in the start phase for scheduling of the observation, and end_margin specifies an allowed tolerance in the end phase for scheduling of the observation. The PHASE parameter can be used in conjunction with the WINDOW parameter to constrain the overall time interval to which the ephemeris applies, and with the REPETITION parameter to specified the number of times the observation is to be scheduled. This is an optional parameter. If the PHASE parameter is specified, only the period and epoch arguments are required.
REPETITION	The REPETITION parameter is used to specify the number and interval for repetitive observations. The REPETITION parameter has three arguments: period specifies the period for the repetition of the observation, delta specifies an allowed tolerance in the period for scheduling of the observation, and number specifies the number of times the observation is to be repeated. If the REPETITION parameter is specified alone, the period and number arguments are required. If the REPETITION parameter is used in conjunction with the PHASE parameter, only the argument number can be supplied; the period and delta arguments will be ignored if provided.
PRECEDING	The PRECEDING parameter specifies a required precedence between observation requests. The PRECEDING parameter has three arguments: reqid specifies the request ID for the request which must precede this observation in scheduling order, minimum lead specifies the minimum time delay between the end of the preceding request and the start of this observation, the maximum lead specifies the maximum time delay between the end of the preceding request and the start of this observation. This is an optional parameter. If PRECEDING is specified, only the reqid argument is required. If the minimum and maximum lead arguments are omitted, no constraints are placed on the timing of the requests, only on the order of the requests. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.

OVERLAP	The OVERLAP parameter specifies requests that must be scheduled concurrently. The OVERLAP parameter has three arguments: reqid specified the request id for the request that must overlap this request, start_lead specifies the maximum delay between the start of this request and the start of the overlapping request, end_lead specifies the maximum delay between the end of the overlapping request and the end of this request. This is an optional parameter. If OVERLAP is specified, only the reqid argument is required. If the start_lead and end_lead arguments are omitted, the entire request duration must be covered by the referenced overlapping request. If both the PRECEDING and OVERLAP parameters are specified, they cannot reference the same reqid.
SEGMENT	The SEGMENT parameter specified whether an observation can be scheduled as separate segments and controls the segmentation of the observation. The SEGMENT parameter has three arguments: max_number specifies the maximum number of segments in which the observation can be scheduled, min_duration specifies the minimum duration of any segment, and max_separation specifies the maximum time separation between any two segments. This is an optional parameter.
MOON	The MOON parameter provides information to override the Moon occultation checks performed during the scheduling of the observation. The MOON parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the Moon to be used in checking for Moon occultation and status is used to turn off the Moon occultation check completely. This parameter is optional. If Moon is provided, either the status or the avoid_angle argument is required. If the MOON parameter or the status sub-parameter is omitted, the status of the Moon occultation check is assumed to be "on" and the occultation check will be performed.
SUN	The SUN parameter provides information to override the Sun occultation checks performed during the scheduling of the observation. The SUN parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the Moon to be used in checking for Sun occultation and status is used to turn off the Sun occultation check completely. This parameter is optional. If Sun is provided, either the status or the avoid_angle argument is required. If the SUN parameter or the status sub-parameter is omitted, the status of the Sun occultation check is assumed to be "on" and the occultation check will be performed.

FSS	The FSS parameter provides information to override the FSS FOV checks performed during the scheduling of the request. The FSS parameter has two arguments: fss_fov_angle specifies the FSS FOV angle in degrees from the boresight of the FSS to be used in checking if the Sun is in the FSS FOV for the requested target, and status is used to turn off the FSS FOV check completely. This parameter is optional. If FSS is provided, only the fss_fov_angle argument is required. If the FSS parameter or the status sub-parameter is omitted, the status of the FSS FOV check is assumed to be "on" and the FSS FOV check will be performed.
EARTH	The EARTH parameter provides information to override the earth occultation checks performed during the scheduling of the observation. The EARTH parameter has two arguments: avoid_angle specifies the avoidance angle in degrees from the limb of the earth to be used in checking for earth occultation and status is used to turn off the earth occultation check completely. This parameter is optional. If EARTH is provided, either the status or avoid_angle argument is required. If the EARTH parameter or the status sub-parameter is omitted, the status of the earth occultation check is assumed to be "on" and the occultation check will be performed.
PLANET	The PLANET parameter provides information to modify the Planet occultation checks performed during the scheduling of the observation. The PLANET parameter has three arguments: id specifies the planet occultation check to be modified, avoid_angle specifies the avoidance angle in degrees from the planet to be used in checking for planet occultation, and status is used to turn off the planet occultation check completely. This parameter is optional. If PLANET is provided, either status or both the id and avoid_angle arguments are required. If the PLANET parameter or the status sub-parameter is omitted, the status of the planet occultation check is assumed to be "on" and the occultation check will be performed.
OBJECT	The OBJECT parameter provides information to modify the celestial object checks performed during the scheduling of the observation. The OBJECT parameter has three arguments: id specifies the celestial object check to be modified and is used as a look-up into the ODB table of celestial object constraints, avoid_angle specifies the avoidance angle in degrees from the celestial object to be used in checking for celestial object interference, and status is used to turn off the celestial object occultation check completely. This parameter is optional. If OBJECT is provided, either status or both the id and avoid_angle arguments are required. If the OBJECT parameter or the status

sub-parameter is omitted, the status of the celestial object check is assumed to be "on" and the check will be performed.

E_RADIATION	The E_RADIATION parameter provides information to modify the radiation zone avoidance checks for electrons performed during the scheduling of the observation. The E_RADIATION parameter has three arguments: status is used to turn off the radiation zone avoidance check for electrons completely, energy is the electron particle energy, and flux is the electron particle flux level above which the observation cannot be scheduled. This parameter is optional. If E_RADIATION is provided, either status or both the energy and flux parameters are required. If the E_RADIATION parameter or the status sub-parameter is omitted, the status of the electron radiation zone avoidance check is assumed to be "on" and the check will be performed.
P_RADIATION	The P_RADIATION parameter provides information to modify the radiation zone avoidance checks for protons performed during the scheduling of the observation. The P_RADIATION parameter has three arguments: status is used to turn off the radiation zone avoidance check for protons completely, energy is the proton particle energy, and flux is the proton particle flux level above which the observation cannot be scheduled. This parameter is optional. If P_RADIATION is provided, either status or both the energy and flux parameters are required. If the P_RADIATION parameter or the status sub-parameter is omitted, the status of the proton radiation zone avoidance check is assumed to be "on" and the check will be performed.
ECLIPSE	The ECLIPSE parameter provides information to control scheduling of the observation during orbit day or orbit night. The ECLIPSE parameter has two arguments: status is used to turn off the spacecraft eclipse check completely, day/night is used to indicate during which portion of the orbit the observation must be scheduled. This parameter is optional. If ECLIPSE is provided, both the status and the day/night parameter is required. If the ECLIPSE parameter or the status sub-parameter is omitted, the status of the eclipse check is assumed to be "on" and the observation will be scheduled in orbit day only.
ALTITUDE	The ALTITUDE parameter specifies an altitude constraint check to be used during scheduling of the request. The ALTITUDE parameter has three arguments: status is used to turn on the altitude check, min_altitude and max_altitude specify the lowest and highest altitudes at which the request can be scheduled, respectively. This parameter is optional. If ALTITUDE is provided with a status of "on", either the min_altitude or max_altitude subparameters are required. If the ALTITUDE parameter or the

status subparameter is omitted, the status of the altitude constraint check is assumed to be "off" and the check will not be performed.

3.24.3.4 HDR Statement Syntax

HDR, HDR_ID=ASCII, [SCH_ALG=Integer] [,TIME_SPAN=(beg_time,end_time)] [,WEIGHT=(class_id,cl_weight)]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	HDR_ID	Observation List Identifier	ASCII String	n/a	1-11 char	
	SCH_ALG	Scheduling algorithm used (1-20).	Integer	n/a	1-20	n/a
	TIME_SPAN					
	beg_time	Beginning time of oversubscribed OR list. (HOSC GMT)	Absolute time	GMT	n/a	n/a
	end_time	Ending time of oversubscribed OR list. (HOSC GMT)	Absolute time	GMT	n/a	n/a
	WEIGHT					
	class_id	Unique class identifier.	ASCII	n/a	1-8 char	
	cl_weight	Weighting factor for this class.	Integer	n/a	1-10	

HDR

The header statement is used to define general information on the schedule such as time span, weighting factors, etc. The allowed HDR parameter definitions are as follows:

HDR_ID — The HDR_ID parameter defines a unique observation list identifier used to track the observation list in the ASC and OFLS systems. The last 3 characters are reserved for a two digit revision indicator in the form “_nn”.

SCH_ALG — The SCH_ALG parameter defines the scheduling algorithm to be used. The argument is an Integer value that defines a particular scheduling algorithm (1=maximize time on target; 2=maximize time on target in priority order, etc.)

TIME_SPAN — The TIME_SPAN parameter is used to specify the time span for the schedule. The TIME_SPAN parameter has two arguments: beg_time is the beginning time of the schedule in HOSC GMT, end_time is the end time of the schedule in HOSC GMT.

WEIGHT — The WEIGHT parameter is used to specify the weighting factor for the various classes. The WEIGHT parameter has two arguments:

classid is the identifier for the class, cl_weight is the weighting factor for that class. Multiple weight parameters can be specified.

3.24.3.5 BEGIN_COMMENT Statement Syntax

BEGIN_COMMENT[,ID=string]

R	*Parameter*	*Description*	*Type*	*Units*	*Range*	*Resolution*
*	ID	Request Identifier	ASCII string	n/a	1-8 char	n/a

BEGIN_COMMENT

The begin comment statement is used to mark the beginning of a comment. All text that follows, until the END_COMMENT statement, will be considered a comment for that ID. The allowed BEGIN_COMMENT parameter definitions are as follows:

ID — The ID parameter provides a unique identifier that is retained and passed to command management for tracing each observation through ground processing, onboard execution, and post-observation data processing. This is a required parameter for comments that are to be passed to command management.

3.24.3.6 END_COMMENT Statement Syntax

END_COMMENT

END_COMMENT

The end comment statement is used to mark the end of a comment. This statement must be used in conjunction with the BEGIN_COMMENT statement. There are no parameters associated with this statement.

3.25 ORBIT EVENTS, DEFINITIVE

3.25.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	OFLS (ISS)
Data Routinely Accessed By:	OFLS (All except AD&SC)
Format Structure:	Binary, Indexed
Delivery Media	Electronic Transfer

3.25.2 CONTENT

The orbit events file provides the times for each calculated orbit event: ascending node, Earth shadow entry and exit, radiation zone entry and exit, and DSN contact LOS and AOS. Each orbit event is a separate record specifying the event type, the time of the event, and current orbit number for the event. The file can be accessed by three indices: the first is

the time (ASCII), the orbit number, and event mnemonic of the event; the second is the time (ASCII) of the event; the third key is the orbit number and the event id; and the fourth is the event ID.

3.25.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Indexed

FILE SIZE (ESTIMATED): 600,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Orbit Event Record

Record Format: Fixed

Record Length: 42 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_oev_real_gmt	Orbit event time (seconds from reference).	Real		8
odb_oev_ASCII_tim	Orbit event time (HOSC GMT).	ASCII		21
odb_oev_orb_num	Orbit number of orbit event.	ASCII		5
odb_oev_id	ID of orbit event. Allowed values are: EASCNCR = ascending node crossing EPERIGEE = perigee time EAPOGEE = apogee time PENTRY = Earth penumbra entry PEXIT = Earth penumbra exit EONIGHT = Earth umbra entry EODAY = Earth umbra exit EEnRADZm = Electron radiation zone energy level n, flux level m entry XEnRADZm = Proton radiation zone energy level n, flux level m exit Egsid = ground station, gsid, acquisition of signal, where gsid may e any of the ground station names psecified in the DSN Characteristics Record of the	ASCII		8

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
	AXAF Characteristics Xgsid = ground station, gsid, loss of signal where gsid may be any of the ground station names specified in the DSN Characteristics Record of the AXAF Characteristics EPnRADZm = Proton radiation zone energy level n, flux level m entry XPnRADZm = Proton radiation zone energy level n, flux level m exit			

3.26 ORBIT EVENTS, PREDICTIVE

3.26.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	OFLS
Data Provided By:	OFLS (ISS)
Data Routinely Accessed By:	OFLS (All except AD&SC)
Format Structure:	Binary, Indexed
Delivery Media	Electronic Transfer

3.26.2 CONTENT

The orbit events file provides the predicted times for each calculated orbit event: ascending node, Earth shadow entry and exit, radiation zone entry and exit, and DSN contact LOS and AOS based on the predictive ephemeris. Each orbit event is a separate record specifying the event type, the time of the event, and current orbit number for the event. The file can be accessed by three indices: the first is the time (ASCII) and event mnemonic of the event, the second is the orbit number of the event and the event ID, and the third is the event ID.

3.26.3 FORMAT DESCRIPTION

The format of each record in the predictive orbit events file is identical to the format for the definitive orbit events file defined in Section 3.25.

3.27 RADIATION ZONE DEFINITIONS

3.27.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC (NSSDC)
Data Provided By:	TRW (NSSDC)
Data Routinely Accessed By:	OFLS (ISS)
Format Structure:	Binary, Sequential
Delivery Media:	Electronic Transfer

3.27.2 CONTENT

This element contains data files used by the National Space Science Data Center (NSSDC) AP-8 and AE-8 radiation models. There are four files in this element: solar minimum and solar maximum data for the electron and proton radiation models. The format of the file for each of the models is identical. The first field in the header record specifies whether the electron or proton model data is for solar minimum or solar maximum. The data is specified by tables of fluxes. The tables are broken into sub-maps for electron or proton energy, and each sub-map is broken into sub-sub-maps for McIlwain value.

3.27.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 85000 (Bytes)

RECORD LAYOUT:

Record Identifier: Radiation Model Header Record

Record Format: Fixed

Record Length: 32(Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rad_ihead(1)	Model type: 1 = AE8MAX, 2 = AE8MIN for electron models; 1 = AP8MAC, 2 = AP8MIC for proton models	Integer		4
odb_rad_ihead(2)	Increments per decade of logarithmic flux.	Integer		4
odb_rad_ihead(3)	Epoch of model.	Integer		4
odb_rad_ihead(4)	Scale factor for energy.	Integer		4
odb_rad_ihead(5)	Scale factor for McIlwain value.	Integer		4

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rad_ihead(6)	Scale factor for B/B0.	Integer		4
odb_rad_ihead(7)	Scale factor for logarithm of fluxes.	Integer		4
odb_rad_ihead(8)	Number of elements in map	Integer		4

RECORD LAYOUT:

Record Identifier: Radiation Model Data Record

Record Format: Fixed

Record Length: 80032(Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rad_map(1)	Number of elements in sub-map.	Integer		4
odb_rad_map(2)	Energy for this sub-map, scaled by the energy scale factor in the header record	Integer		4
odb_rad_map(3)	Number of elements in sub-sub-map.	Integer		4
odb_rad_map(4)	L-value for this sub-sub-map, scaled by the McIlwain scale factor in the header record.	Integer		4
odb_rad_map(5)	Logarithm of flux at equator, scaled by the logarithm of flux scale factor in the header record.	Integer		4
odb_rad_map(6)	Logarithm of flux at the second step (step size determined by the increments per decade of logarithmic flux in the head record)	Integer		4
odb_rad_map(7)	Logarithm of flux at the third step (step size determined by the increments per decade of logarithmic flux in the head record)	Integer		4
	...logarithm of flux data repeats for each element in the sub-sub-map	Integer		
odb_rad_map(odb_rad_map(3))	Logarithm of flux at the last step (corresponding to the number of elements in the sub-sub-map)	Integer		4
odb_rad_map(odb_rad_map(3) + 1)	Number of elements in next sub-sub-map	Integer		4

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
	... data in each sub-sub-map repeats fields 3 to odb_rad_map(3) for each sub-sub-map	Integer		
odb_rad_map(odb_rad_ map(1))	Last element in sub-map	Integer		4
odb_rad_map(odb_rad_ map(1) + 1)	Number of elements in next sub-map	Integer		4
	... data in each sub-map repeats fields 1 to odb_rad_map(1) for each sub-map			
odb_rad_map (odb_rad_ihead(8))	Last element in last sub-sub-map in last sub-map	Integer		4

3.28 RELATIVE TIME SEQUENCE

3.28.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: TRW, ASC

Data Routinely Accessed By: OFLS (CM)

Format Structure: ASCII, Sequential

Delivery Media: Electronic transfer

3.28.2 CONTENT

The relative time sequence definition element is used by command management (CM) to expand relative time sequence references in the DOT, FOT, or command sequence requests into sequences of spacecraft commands. Relative time sequences are loaded onboard once, and are executed from the absolute time sequence or another relative times sequence. Each file in the element defines a relative sequence. The text in the following section describes the format and syntax for relative time sequences and how they are used by CM.

3.28.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 150,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Command Sequence Definition Record

Record Format: Fixed

Record Length: 80 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
cmd_seq_text	Text line containing command sequence definitions	ASCII		80

3.28.3.1 Relative Time Sequence Statement Definitions

The following provide the format and contents of the statements allowed within a relative time sequence. The relative time sequence request consists of a sequence type specification, the command sequence mnemonic, and the associated parameter lists. All relative time sequences invoked will be contained in the relative time sequence or command sequence definition elements of the ODE. The statements allowed in a relative time sequence are: RTS, SIMPKT, ACAPKT, EIOPKT, /CMD, IF, ELSE, _ENDIF, SET, _DELETE, AON, AOF, and ACIS. The statements follow the same conventions and syntax as those described in Section 3.7 for command sequence definitions.

3.29 SCHEDULED OR/ER DATA

3.29.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (MPS)

Data Routinely Accessed By: ASC

Format Structure: ASCII

Delivery Media: Electronic Transfer

3.29.2 CONTENT

The scheduled OR/ER data element comprises two types of data files: scheduled OR/ER data file and scheduled OR/ER comment files. The Scheduled OR/ER data file contains information on scheduled OR/ERs including: general information on the schedule (time span, total observing time, scheduling objective used, etc.), detailed information for each OR/ER (start and end times, id, associated acquisition stars, guide stars, and fid lights, maneuver data, visibility data, etc.), idle period information, SSR/communication information and a reference comment.

There is a separate record for each OR and ER; separate records for references to each comment associated with an OR or ER; and additional records for each idle time interval; each playback, communications, momentum dump, telemetry, and activity request; and

each visibility event. There are no line feed characters terminating records in the scheduled OR/ER data file.

Each scheduled OR/ER comment file contains a free form comment referenced by an OR/ER. This format allows multiple separate comments for each scheduled OR/ER without limiting the size of the comment.

3.29.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 150,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Header Record

Record Format: Fixed

Record Length: 490 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_type	Record type identifier; allowed value is "HDR"	ASCII		3
odb_sched_id	Defines unique schedule ID.	ASCII		8
odb_prev_sched_id	Identifies unique schedule id for the previous schedule used for continuity at the start of the current schedule; value will be “INITIAL” if no previous schedule exists	ASCII		8
odb_orlist_id	Defines unique observation request list id from which the schedule generated	ASCII		11
odb_erlist_id	Defines unique engineering request list ids from which the schedule was generated (index = engineering list)	ASCII	10	110
odb_sched_obj	Defines scheduling objective used. (1 to 20) (1=maximize time on target, 2=maximuize priority time on target)	Integer		2
odb_start_time	Defines start time of schedule. (HOSC GMT)	ASCII		21
odb_end_time	Defines end time of schedule. (HOSC GMT)	ASCII		21
odb_sched_time	Defines the total schedule duration (relative time)	ASCII		17

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_or_list_name	Defines name of OR List from the ODE used to generate this schedule	ASCII		25
odb_or_list_update	Defines last update time of the OR list used to generate this schedule (HOSC GMT)	ASCII		21
odb_tot_obs_time	Defines total observing time of schedule (relative time)	ASCII		17
odb_tot_src_time	Defines total observing time on source (excluding time exceeding requested time)	ASCII		17
odb_tot_add_time	Defines total observing time exceeding requested observation duration (relative time)	ASCII		17
odb_frac_add_time	Defines fraction of observing time exceeding requested observation duration	Real		10
odb_tot_sub_time	Defines total observing time below requested observation duration (relative time)	ASCII		17
odb_frac_sub_time	Defines fraction of observing time below requested observation duration	Real		10
odb_tot_slew_time	Defines total slew time of schedule (relative time).	ASCII		17
odb_frac_slew_time	Defines the fraction of the total schedule duration during which the spacecraft is maneuvering	Real		10
odb_tot_acq_time	Defines total acquisition time of schedule (relative time)	ASCII		17
odb_frac_acq_time	Defines the fraction of the total schedule duration during which the spacecraft is acquiring acquisition and guide stars	Real		10
odb_tot_idle_time	Defines total idle time of schedule (relative time)	ASCII		17
odb_frac_idle_time	Defines the fraction of the total schedule duration during which the spacecraft is idle	Real		10
odb_tot_rad_time	Defines total radiation time of schedule (relative time)	ASCII		17

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_frac_rad_time	Defines the fraction of the total schedule duration during which the spacecraft is in the radiation zone	Real		10
odb_tot_ecl_time	Defines total time the Sun is eclipsed by the Earth (relative time)	ASCII		17
odb_frac_ecl_time	Defines the fraction of the total schedule duration during which the Sun is eclipsed by the Earth	Real		10
odb_av_efficiency	Defines the schedule average efficiency	Real		10
odb_total_gas_used	Defines total gas used (liters)	Real		10

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Data Record

Record Format: Fixed

Record Length: 3363 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_id	Record type identifier; allowed values are “OBS” or “CAL”	Real		3
odb_target_quat	Defines the target quaternion up-linked to the spacecraft OR/ER	Real	4	40
odb_sstart_ang	Defines sun angle at start time of OR/ER.	Real		10
odb_send_ang	Defines sun angle at end time of OR/ER.	Real		10
pdb_sclose_ang	Defines the angle of the closest approach between the spacecraft x-axis and the center of the Sun during the OR/ER time span. (degrees)	Real		10
pdb_estart_ang	Defines the angle between the spacecraft x-axis and the center of the Earth at the OR/ER start time. (degrees)	Real		10
pdb_eend_ang	Defines the angle between the spacecraft x-axis and the center of the Earth at the OR/ER end time. (degrees)	Real		10
pdb_eclose_ang	Defines the angle of the closest approach between the spacecraft x-axis and the center of the Earth during the OR/ER time span. (degrees)	Real		10
pdb_mstart_ang	Defines the angle between the spacecraft x-axis and the center of the Moon at the OR/ER start time. (degrees)	Real		10
pdb_mend_ang	Defines the angle between the spacecraft x-axis and the center of the Moon at the OR/ER end time. (degrees)	Real		10

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
pdb_mclose_ang	Defines the angle of the closest approach between the spacecraft x-axis and the center of the Moon during the OR/ER time span. (degrees)	Real		10
pdb_pstart_ang	Defines the angle between the spacecraft x-axis and an x-ray object at the OR/ER start time. The values are provided in the same order as odb_pclose_id. (degrees)	Real	*20*	*200*
pdb_pend_ang	Defines the angle between the spacecraft x-axis and an x-ray object at the OR/ER end time. The values are provided in the same order as odb_pclose_id. (degrees)	Real	20	200
pdb_pclose_ang	Defines the angle of closest approach between the spacecraft x-axis and an x-ray object during the OR/ER timespan. The values are provided in the same order as odb_pclose_id. (degrees)	Real	20	200
pdb_kstart_ang	Defines the angle between the spacecraft x-axis and an x-ray object at the OR/ER start time. The values are provided in the same order as odb_kclose_id. (degrees)	Real	20	200
pdb_kend_ang	Defines the angle between the spacecraft x-axis and an x-ray object at the OR/ER end time. The values are provided in the same order as odb_kclose_id. (degrees)	Real	20	200
pdb_kclose_ang	Defines the angle of closest approach between the spacecraft x-axis and an x-ray object during the OR/ER timespan. The values are provided in the same order as odb_kclose_id. (degrees)	Real	20	200
odb_acq_stars	Defines acquisition stars associated with OR/ER (index: 1 = Y component and Z component of the image unit vector in the ECI frame., 2= image number)	Real	2x8	160

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_guide_images	Defines guide stars, fid lights, or monitor windows associated with OR/ER (index: 1 = star or fid light indicator with values =1, guide star, =2, fid light, = 3, monitor window; Y component of the image unit vector in the ECI frame; Z component of the image unit vector in the ECI frame, 2= image number)	Real	3x8	240
odb_fom	Figure of merit for guide stars associated with OR/ER (arcseconds2)	Real		10
odb_roll_ang	Defines roll angle for this instance of the OR/ER. (degrees)	Real		10
odb_slew_ang	Defines the slew angle from the previous target attitude to the target attitude for this instance of the OR/ER. (degrees)	Real		10
odb_instance_num	Defines instance number of OR/ER.	Integer		2
odb_req_id	Defines OBS or CAL request ID	ASCII		8
odb_obs_id	Defines observation ID uplinked to the spacecraft	Integer		5
odb_acq_id	Defines AGASC ids for acquisition stars associated with OR/ER. If id is not available, field will be blank. (index: image number)	Integer	8	80
odb_guide_id	Defines AGASC ids for guide stars or monitor windows associated with OR/ER; defines the fid light number for fid lights associated with OR/ER. If id is not available for stars or monitor windows, field will be blank. (index: image number)	Integer	8	80
odb_obs_start_time	Defines start time of this instance of the OR/ER in HOSC GMT.	ASCII		21
odb_obs_end_time	Defines end time of this instance of the OR/ER in HOSC GMT.	ASCII		21
odb_obs_dur	Defines total duration of observation up to nominal requested duration (no time exceeding requested) (relative time)	ASCII		17

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_obs_dur_extra	Defines observation time exceeding requested duration (relative time)	ASCII		17
odb_mstart_time	Defines start time of maneuver for this instance OR/ER. (HOSC GMT) (If no maneuver for this instance, mstart = 0)	ASCII		21
odb_mend_time	Defines end time of maneuver for this instance OR/ER. (HOSC GMT) (If no maneuver for this instance, mend = 0)	ASCII		21
odb_trans_stime	Defines SIM transition start time for this OR/ER. (If no transition for this instance. odb_trans_stime = 0). (HOSC GMT)	ASCII		21
odb_trans_etime	Defines SIM transition end time for this OR/ER. (If no transition for this instance. odb_trans_stime = 0). (HOSC GMT)	ASCII		21
odb_trans_wstime	Defines warm-up start time for this OR/ER. (If no transition for this instance. odb_trans_stime = 0). (HOSC GMT)	ASCII		21
odb_trans_wetime	Defines warm-up end time for this OR/ER. (If no transition for this instance. odb_trans_stime = 0). (HOSC GMT)	ASCII		21
odb_sclose_time	Defines the time of closest approach between the spacecraft x-axis and the Sun during the OR/ER time span. (HOSC GMT)	ASCII		21
odb_eclose_time	Defines the time of closest approach between the spacecraft x-axis and the Earth during the OR/ER time span. (HOSC GMT)	ASCII		21
odb_mclose_time	Defines the time of closest approach between the spacecraft x-axis and the Moon during the OR/ER time span. (HOSC GMT)	ASCII		21
odb_pclose_time	Defines the time of closest approach between the spacecraft x-axis and an x-ray object during the OR/ER time span. The values are provided in the same order as odb_pclose_id. (HOSC GMT)	ASCII	20	420

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_pclose_id	Ids for the X-Ray objects specified by object position as defined in the ODE_Attitude_Hold Constraints record variable odb_pos_obj_id	ASCII	20	160
odb_kclose_time	Defines the time of closest approach between the spacecraft x-axis and an x-ray object during the OR/ER time span. The values are provided in the same order as odb_kclose_id. (HOSC GMT)	ASCII	20	420
odb_kclose_id	Ids for the X-Ray objects specified by Keplerian orbital elements as defined in the ODE_Attitude_Hold Constraints record variable odb_kepl_obj_id	ASCII	20	160

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Visibility Record

Record Format: Fixed

Record Length: 40 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_type	Record type identifier; allowed values is “VIS”	ASCII		3
odb_obs_id	Defines observation ID.	ASCII		8
odb_vis_type	Defines type of visibility constraint (the first character indicates entry (‘E’) or exit (‘X’), characters 2 - 7 = ‘EOCULT’ indicate an Earth occultation event, characters 2 - 7 = ‘MOCLT’ indicate an Moon occultation event, characters 2 - 8 = as specified in the Orbit Events data element indicate a radiation zone event)	ASCII		8
odb_event_time	Defines time of visibility event for this OR/ER. (Extended HOSC GMT)	ASCII		21

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Idle Period Record

Record Format: Fixed

Record Length: 45 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_type	Record type identifier; allowed values is “IDL”	ASCII		3
odb_istart_time	Defines the start time of an idle period. (HOSC GMT)	ASCII		21
odb_iend_time	Defines the end time of an idle period. (HOSC GMT)	ASCII		21

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER SSR Support Record

Record Format: Fixed

Record Length: 45 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_type	Record type identifier; allowed values are "PBK", "COM", "MOM", "TLM", or "ACT"	ASCII		3
odb_start_time	Defines start time of playback, communications, momentum management, telemetry, or generic activity request (HOSC GMT)	ASCII		21
odb_end_time	Defines end time of playback, communications, momentum management, telemetry, or generic activity request (HOSC GMT)	ASCII		21

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Comment Record

Record Format: Fixed

Record Length: 91 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_rec_type	Record type identifier; allowed values are "CMT" or "ERR"	ASCII		3

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_obs_id	Defines observation ID.	ASCII		8
odb_comment_file	Defines filename containing comments.	ASCII		80

3.29.3.1 Comment Format Description

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 400 (Bytes)

RECORD LAYOUT:

Record Identifier: Scheduled OR/ER Comment File Record

Record Format: Fixed

Record Length: 400 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_comment_line	Defines one line of a free form comment.	ASCII		80

3.30 SENSOR CALIBRATION DATA

3.30.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (AD&SC)

Data Routinely Accessed By: OFLS (AD&SC)

Format Structure: Binary, Indexed

Delivery Media Electronic Transfer

3.30.2 CONTENT

The Sensor Calibration Data File contains the calibration parameters calculated by the OFLS for the FSS, IRU, and AC (TBR). The Sensor Calibration Data File is formatted as an indexed file with two indices: the first is the time in GMT of the calibration concatenated with the sensor ID, the second is the time in GMT of the calibration.

3.30.3 FORMAT DESCRIPTION

FILE SIZE (ESTIMATED): 3,000 (Bytes)

FILE ORGANIZATION: Indexed

RECORD LAYOUT:

Record Identifier: FSS Calibration Record

Record Format: Fixed

Record Length: 241 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_fss_alpha_coeffs	Calibration coefficients for the angular measurement about the FSS y-axis	Real	9	72
odb_fss_beta_coeffs	Calibration coefficients for the angular measurement about the FSS x-axis	Real	9	72
odb_fss_ misalign	Misaligned FSS frame to Nominal FSS frame rotational transformation matrix	Real	3x3	72
odb_fss_id	Identifies the FSS to which this record applies. (‘FSS1’ = FSS 1, ‘FSS2’ = FSS 2). In this context FSS implies two measurement axes.	ASCII		4
odb_fss_gmt	Time of the FSS calibration in HOSC GMT	ASCII		21

RECORD LAYOUT:

Record Identifier: IRU Calibration Record

Record Format: Fixed

Record Length: 145 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_iru_scale_fact_align	IRU scale factor alignment matrix for a single channel combination (degrees/count)	Real	3x4	96
odb_iru_drift_rate	IRU drift rate bias vector for a single channel combination (degrees/second)	Real	3	24

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_iru_channel_comb	Channel configuration identifier (indicates to which combination of three or four of the eight IRU channels the data in this record applies)	Integer		4
odb_iru_gmt	Time of IRU calibration record in HOSC GMT	ASCII		21

3.31 SOFTWARE UPDATES, AC

3.31.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: TRW

Data Provided By: SMF

Data Routinely Accessed By: OFLS (CM)

Format Structure: Refer to SMF/OCC ICD for format definition

Delivery Media: Electronic Transfer

3.31.2 CONTENT

The AC Software Updates data element contains software updates to the ACA. The updates are generated by the Software Maintenance Facility (SMF) and packaged by the OFLS into Command Load for uplink.

3.31.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.32 SOFTWARE UPDATES, ACIS

3.32.1 HEADER INFORMATION

Element Type: ODE

Format provided by: ASC

Data provided by: ASC

Data routinely accessed by: OFLS (CM)

Format structure: ASCII Flat File

Delivery media: Electronic transfer from the ASC data system.

3.32.2 CONTENT

This element contains ACIS executable code, for use by the OFLS in generating ACIS flight software update loads for uplink to the spacecraft. The executable code consists of patches to the ACIS BEP main software. Header information is included with each patch, that is used by the OFLS in assembling the patches into command loads. These patches will be provided by the ASC and will not be edited once they are in the database. New patches may be provided throughout the life of the mission, probably more-frequently at the beginning of the mission than at the end.

3.32.3 FORMAT DESCRIPTION

This element is a flat file made up of a sequence of records, each with header data (Table 3-11) and a binary patch (Table 3-12) to the BEP software, in the following format (preliminary; to be finalized by the ACIS project):

Table 3-11 ACIS S/W Update Format - HEADER

Patch identification tag	Integer
Patch-start address in ACIS BEP memory	Binary
Length of the patch	Integer
Annotations and notes, describing the patch	256 char

Table 3-12 ACIS S/W Update Format - DATA

Binary data	256 Kbytes

Patch size is not fixed. Typical size is 256 Kbytes. Number of patches is not expected to exceed (TBD by the ASC and ACIS project).

3.33 SOFTWARE UPDATES, CPE

3.33.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.33.2 CONTENT

The CPE Software Updates data element contains software updates to the 2 CPE's. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.33.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.34 SOFTWARE UPDATES, CTU EEPROM

3.34.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.34.2 CONTENT

The CTU EEPROM Software Updates data element contains updates to the 2 CTU EEPROM's. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.34.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.35 SOFTWARE UPDATES, I-EPHIN

3.35.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.35.2 CONTENT

The I-EPHIN Software Updates data element contains updates to the I-EPHIN. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.35.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.36 SOFTWARE UPDATES, IU EEPROM

3.36.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.36.2 CONTENT

The IU EEPROM Software Updates data element contains the updates to the IU EEPROM's. Each IU EEPROM contains 2 programs. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.36.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.37 SOFTWARE UPDATES, OBC

3.37.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.37.2 CONTENT

The OBC Software Updates data element contains software to the OBC. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.37.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.38 SOFTWARE UPDATES, SIM

3.38.1 HEADER INFORMATION

Element Type:	ODE

Format Provided By:	TRW
Data Provided By:	SMF
Data Routinely Accessed By:	OFLS (CM)
Format Structure:	Refer to SMF/OCC ICD for format definition
Delivery Media:	Electronic Transfer

3.38.2 CONTENT

The SIM Software Updates data element contains Software Updates to the SIM. The updates are generated by the SMF and packaged by the OFLS into Command Load for uplink.

3.38.3 FORMAT DESCRIPTION

Refer to SMF/OCC ICD for format definition.

3.39 SOLAR, LUNAR, PLANETARY DATA

3.39.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	CSC
Data Provided By:	JPL
Data Routinely Accessed By:	OFLS (All except CM)
Format Structure:	Binary, Direct Access
Delivery Media	Electronic Transfer

3.39.2 CONTENT

The SLP file provides position and velocity information for the Sun, the Moon, and the Planets. The file contains blocks of Chebyshev coefficients of position and the coefficients needed to interpolate each block for the SLP ephemerides.

3.39.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Direct

FILE SIZE (ESTIMATED): 4,000,000 (Bytes)

RECORD LAYOUT:

Record Identifier: SLP Record 1

Record Format: Fixed

Record Length: 2856 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_slp_ttl	Title in the JPL SLP file.	ASCII	14x3	252
odb_slp_cnam	Array of constant names.	ASCII	400	2400
odb_slp_ss	Time span of the SLP ephemeris file (index: 1 = Starting JED of the ephemeris file. 2 = Ending JED of the ephemeris file. 3 = Number of days covered by each block of Chebyshev coefficients).	Real	3	24
odb_slp_ncon	Number of constants.	Integer		4
odb_slp_au	Number of kilometers per astronomical unit.	Real		8
odb_slp_emrat	Earth-Moon mass ratio used in generating the data.	Real		8
odb_slp_l	Pointers into the data buffers (SLP record 3) needed by interpolation routine. L(1,I) = Position in buffer of coefficients for body I. L(2,I) = Number of coefficients per component. L(3,I) = Number of sets of coefficients in full array. where I = body number: 1 = Mercury, 2 = Venus, 3 = Earth-Moon barycenter, 4 = Mars, 5 = Jupiter, 6 = Saturn, 7 = Uranus, 8 = Neptune, 9 = Pluto, 10 = Moon, 11 = Sun	Integer	(3,12)	144
odb_slp_denum	Planetary ephemeris number.	Integer		4
odb_slp_lpt	Pointers needed by interpolation routine.	Integer	3	12

RECORD LAYOUT:

Record Identifier: SLP Record 2

Record Format: Fixed

Record Length: 3200 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_slp_cval	Array of constant values corresponding to constant names in odb_slp_ncon in SLP record 1.	Real	400	3200

RECORD LAYOUT:

Record Identifier: SLP Record 3

Record Format: Fixed

Record Length: 8000 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_slp_buf	BUF(L(1,I)) = Array of DP Chebyshev coefficients of position. (index: L array defined in SLP record 1).	Real	2000	8000

3.40 SPACECRAFT CLOCK CORRELATION

3.40.1 HEADER INFORMATION

Element Type: ODE

Format Provided By: CSC

Data Provided By: OFLS (SS&EA)

Data Routinely Accessed By: OFLS (All)

Format Structure: Indexed

Delivery Media Electronic Transfer

3.40.2 CONTENT

The Spacecraft Clock Correlation File contains the clock correlation parameters calculated by the OFLS. The correlation parameters are used to convert between GMT and onboard VCDU clock counts. The Spacecraft Clock Correlation File is formatted as an indexed file with one index, the calculated time of the correlation parameters in negative seconds from reference.

3.40.3 Format Description

FILE SIZE (ESTIMATED): 3,000 (Bytes)

FILE ORGANIZATION: Indexed

RECORD LAYOUT:

Record Identifier: Spacecraft Clock Correlation Record

Record Format: Fixed

Record Length (Maximum): 169 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_clock_adj_data_l	First DSN ground receipt time adjusted for delays (seconds form base reference)	Real		8
odb_clock_adj_data_f	Final DSN ground receipt time adjusted for delays (seconds from base reference)	Real		8
odb_clock_ref_cnts	Reference spacecraft clock counts (counts)	Real		8
odb_clock_ref_gmt	Reference GMT computed by least squares fit (seconds from base reference)	Real		8
odb_clock_std_dev	Std deviations of fit term for reference GMT, clock rate, and drift rate; index: 1=GMT, 2=clock rate, 3=drift rate, 4=deft deriv	Real	4	32
odb_clock_rate	Rate of clock (seconds/count)	Real		8
odb_clock_drift	Drift rate of clock (sec/count**2)	Real		8
odb_clock_der_drift	First derivative of clock drift rate (seconds/count^3)	Real		8
odb_clock_errtime	Time at which 0.1, 1.0, and 10.0 millisecond errors develop based on the computed clock correlation parameters; index: 1= 0.1, 2=1.0, 3=10.0 (seconds from base reference)	Real	3	24
odb_clock_variance	Variance of residuals (sec^2)	Real		8
odb_clock_majfm_cnt	Major frame fiducial point (counts)	Integer		4
odb_clock_majfm_utc	Major frame counts in seconds from base reference	Real		8
odb_clock_key	Time of the clock correlation (negative seconds from base reference	Integer		4
odb_clock_key_char	ASCII representation of the time of clock correlation key (format is a C followed by the negative seconds from base reference; C-nnnnnnnn)	ASCII		11

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_clock_base_ref	Base reference time (HOSC GMT)	ASCII		21

3.41 STAR CATALOG

3.41.1 HEADER INFORMATION

Element type: ODE

Format provided by: ASC, CSC

Data provided by: ASC

Data routinely accessed by: OFLS (MPS, AD&SC, SS&EA)

Format structure: FITS format

Delivery media: CD ROM

CD-ROMs (set of four) for initial delivery and bulk replacement; electronic transfer from the ASC data system for small-scale updates. The AXAF Guide and Acquisition Star Catalog (AGASC) data element is delivered as a set of five supporting files containing physical format, organizational and descriptive information, plus a sequential set of about 10,000 FITS Binary files containing the actual star catalog data.

FITS files will be organized in a tree under one root directory for each CDROM. An introductory file (readme.txt) and the supporting tables are duplicated on all discs.

"README.TXT" - is an introductory text file (ASCII) providing a description of the AXAF Star Catalog, its release history, changes in the current version, explanation of selected or new entries in the catalog data, and a brief description of the organization of the data files.

"TABLES" - is a directory containing four text files that describe the global format of the star parameter data:

1. COMMENTS.TXT - Introduction and general comments (ASCII). Describes the organization of the data files, a summary of the AGASC format, including header and star data information, and a description of the history and construction of the catalog.

2. REGIONS - Boundaries of GSC regions (FITS binary table). All information needed to associate a point on the sky with a region file is given in file REGIONS. However, files LR_REG_X, which is an index to the large regions, and SM_REG_X, which is an index to the small regions, are provided to support a more rapid search algorithm based on the geometric arrangement of the regions and their numbering conventions.

3. LG_REG_X - Index to large regions (FITS binary table). This file gives, for each declination band DEC_CTR, the number of the first large region F_LG_REG in the band and the number of large regions N_LG_REG therein.

4. SM_REG_X - Index to small regions (FITS binary table). This file gives, for each large region LG_REG_N, the number F_SM_REG of the first small region and the level of subdivision of the large region DEPTH.

"AGASC" - is a directory containing a sequential set of star files, in FITS Binary table format. Each file corresponds to a spatial cell containing of the order of 2000 data records. The files are organized by declination band, according to the structure defined in REGIONS.TBL. The physical format of each star record within a cell (regions file) is defined in COMMENTS.TBL.

Limited updates to AGASC, primarily for correction of errors, or inclusion of new stars or star data determined after initial delivery, will be made by replacement of individual FITS Binary table files. Corresponding text edits of the descriptive text files may accompany these updates.

3.41.2 CONTENT

AGASC defines positions, proper motions, magnitudes, colors, and their related uncertainties for a set of about 19 million celestial objects. The catalog information is used for selecting acquisition and guide stars during the mission scheduling process, and for star pattern matching during post-facto fine attitude determination and aspect camera calibration processing.

During mission scheduling, the OFLS uses information in the star catalog, together with information in the ORs and selection parameters in the Star Selection Algorithm Parameters (listed as part of the Characteristics in section 3.2), to select acquisition stars and guide/aspect stars for each observation. Specific parameters are provided for each star in the star catalog in order to support this selection process.

The initial version of the AGASC (1.0) was prepared using star data from the Hubble Space Telescope Guide Star Catalog, with additional placeholder fields defined for additional parameters required for AXAF star selection that were to be provided in later releases of the catalog. In version 1.1 of AGASC, color and proper motion information was ingested from the Positions and Proper Motions Catalog (PPM). New entries and placeholder fields were also defined.

A README text file is provided in the AXAF Star Catalog data element to describe the format of the individual star parameter fields in each release. Section 3.41.3 below describes the physical format of the individual star parameter fields that will be in the baseline (final pre-launch) release.

The star parameter data records are organized into spatial cells defined by right ascension (RA) and declination (Dec) intervals. Each cell constitutes a single file. The cells themselves are organized into segments consisting of all cells that make up a declination band. A directory in the data element defines the spatial bounds (RA and Dec) for each cell in each band.

The star catalog data element contains the following information for each star in the catalog:

Star identifier

Celestial position, position uncertainty, and source catalog code

Epoch of position, proper motion, and p.m. source catalog code

Estimated Aspect Camera magnitude, and uncertainty

Morphological class code

Published magnitude, uncertainty, spectral band for which the magnitude is provided, and source catalog code

Two colors (magnitude differences in specified bands), with their uncertainties, and source catalog code

Object variability code, and source catalog code

Quality codes (for use in acquisition and aspect/guide star selection)

Unique star ID numbers for cross-referencing from any merged catalog. For instance XXXXYYYY, where XXXX will be the region number of the original HST GSC file, and YYYY will be the original HST GSC star ID within that file.

3.41.3 FORMAT DESCRIPTION

FILE ORGANIZATION: ASCII, FITS format

FILE SIZE (ESTIMATED): 500,000 (Bytes)

RECORD LAYOUT:

Record Identifier: AXAF Star Catalog Region Record

Record Format: Fixed

Record Length: 48 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
reg_num	Defines region number - corresponds to a guide star file.	Integer		5
spare	Unused			2
reg_ra_low_hr	Defines right ascension of “left” side of the region (hours).	Integer		2
spare	Unused			1
reg_ra_low_mn	Defines right ascension of “left” side of the region (minutes).	Integer		2
spare	Unused			1
reg_ra_low_s	Defines right ascension of “left” side of the region (seconds).	Real		5
spare	Unused			1

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
reg_ra_hi_hr	Defines right ascension of “right” side of the region (hours).	Integer		2
spare	Unused			1
reg_ra_hi_mn	Defines right ascension of “right” side of the region (minutes).	Integer		2
spare	Unused			1
reg_ra_hi_s	Defines right ascension of “right” side of the region (seconds).	Real		5
spare	Unused			1
reg_dec_lo_si	Defines declination of “lower” side of the region (sign).	ASCII		1
reg_dec_lo_dg	Defines declination of “lower” side of the region (degrees).	Integer		2
spare	Unused			1
reg_dec_lo_mn	Defines declination of “lower” side of the region (minutes).	Real		4
spare	Unused			1
reg_dec_hi_si	Defines declination of “upper” side of the region (sign).	ASCII		1
reg_dec_hi_dg	Defines declination of “upper” side of the region (degrees).	Integer		2
spare	Unused			1
reg_dec_hi_mn	Defines declination of “upper” side of the region (minutes).	Real		4

FILE ORGANIZATION: Binary, FITS format

FILE SIZE (ESTIMATED): 2,200,000,000 (Bytes)

RECORD LAYOUT:

Record Identifier: AXAF Star Catalog Identification Record

Record Format: Fixed

Record Length : 104 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
sc_agasc_id	identifying number of star catalog object	Integer		4

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
sc_ra	right ascension (equinox 2000.0, decimal degrees)	Real		8
sc_dec	declination (equinox 2000.0, decimal degrees)	Real		8
sc_pos_err	position error (milli-arcsec)	Integer		2
sc_pos_catid	identifier of star catalog from which position data originated. 0 = No associated catalog 1 = GSC1.1 (Hubble) 2 = PPM (positions and proper motions) 3 = TOC (TYCHO output catalog) 4 = undetermined 5 = undetermined	Integer		1
sc_epoch	Epoch date of ra & dec for objects with proper motion data, epoch is 2000.0 and ra and dec have been updated to epoch J2000, otherwise, it is the epoch of the ra & dec measurement	Real		4
sc_pm_ra	right ascension proper motion (milli-arcsec per year)	integer		2
sc_pm_dec	declination proper motion (milli-arcsec/year)	integer		2
sc_pm_catid	identifier of star catalog from which proper motion data originated (see sc_pos_catid for possible values)	integer		1
sc_mag_aca	magnitude calculated in AXAF ACA bandpass from visual magnitude and color data. Assumes (B-V) = 0.7 when color data is not available	real		4
sc_mag_aca_err	ACA magnitude error (hundredths of magnitude)	integer		2
sc_class	morphological/multiple system code (first five are from GSC): 0 = Star 1 = Galaxy 2 = Blend or member of incorrectly resolved blend 3 = Non-Star 5 = Potential artifact 6 = known multiple system (HIPPARCOS)	integer		2
sc_mag	original magnitude as listed	real		4

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
sc_mag_err	magnitude error (hundredths of magnitude)	Integer		2
sc_mag_band	Defines spectral band code.	Integer		2
sc_mag_catid	identifier of catalog from which magnitude (mag) originated (see sc_pos_catid above)	integer		1
sc_color1	cataloged or estimated B-V color used for sc_mag_aca	Real		4
sc_color1_err	error in sc_color1 (hundredths of magnitude)	Integer		2
sc_c1_catid	identifier of catalog from which sc_color1 originated (see sc_pos_catid above)	Integer		1
sc_color2	second cataloged color	Real		4
sc_color2_err	error in sc_color2 (hundredths of magnitude)	Integer		2
sc_c2_catid	identifier of catalog from which sc_color2 originated (see sc_pos_catid above)	Integer		1
sc_var	variability codes (following HIPPARCOS): Blank = the star is not a known or suspected variable 1 = suspected variable, with a suspected variation of <2 mag 2 = suspected variable, with a suspected variation of >2 mag 3 = known variable, with a suspected variation of >0.2 mag 4 = known variable, with a large amplitude (>2 mag) for which an ephemeris was necessary 5 = known variable with a variation of <0.2 mag	integer		2
sc_var_catid	identifier of catalog from which variability codes originated (see sc_pos_catid above)	integer		1

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
sc_aspq1	Spoiler quality-code 1 for aspect stars: 0 = no neighbors within <=20” (arc sec) 1 = spoiler within 15 < rotation angle <=20” with mag difference >= 4 2 = spoiler within 15 < rotation angle <=20” with mag difference < 4 3 = spoiler with rotation angle <=5” with mag difference >= 4 4 = spoiler within 5 < rotation angle <= 15” with mag difference >= 4 5 = spoiler with rotation angle <= 5” with mag difference < 4 6 = spoiler within 5 < rotation angle <= 15” with mag difference < 4 7 = spoiler with rotation angle <=20” with mag difference <=1 or multiple spoilers	Integer		2
sc_aspq2	Spoiler quality-code 2 for aspect stars 0 = star has no proper motion or PM < 0.5”/year 1 = star has PM >= 0.5"/year	Integer		2
sc_aspq3	Spoiler quality-code for 3 aspect stars indicates that there is >=1 spoiler within r<378” of object that either has very uncertain positions, or is very extended	Integer		2
sc_acqq1	Spoiler quality-code 1 for acquisition stars for slew range 0 < slew angle <= 30 deg. The magnitude difference (in ACA bandpass, hundredths of magnitude) between the brightest star within a radius of [1 + sqrt(2)]*(1/6)*133” around this agasc star. Mag difference is: mag_spoofer = mag_agasc_star	Integer		2
sc_acqq2	same as above except for slew range: 30 < slew angle <= 60 deg and radius around star is : [1 + sqrt(2)]*(2/6)*133”	Integer		2

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
sc_acqq3	same as above except for slew range: 60 < slew angle <= 90 deg and radius around star is: [1 + sqrt(2)]*(3/6)*133”	Integer		2
sc_acqq4	same as above except for slew range: 90 < slew angle <= 120 deg and radius around star is: [1 + sqrt(2)]*(4/6)*133”	Integer		2
sc_acqq5	same as above except for slew range: 120 < slew angle <= 150 deg and radius around star is: [1 + sqrt(2)]*(5/6)*133”	Integer		2
sc_acqq6	same as above except for slew range: 150 < slew angle <= 180 deg and radius around star is: [1 + sqrt(2)]*(6/6)*133”	Integer		2
sc_xref_id1	star id number from catalog 1 (GSC1.1)	integer		4
sc_xref_id2	star id number from catalog 2 (PPM)	integer		4
sc_xref_id3	star id number from catalog 3 (TOC)	integer		4
sc_xref_id4	star id number from catalog 4 (Undetermined)	integer		4
sc_xref_id5	star id number from catalog 5 (undetermined)	integer		4

NOTE: Any entry = -9999 implies a dummy entry, i.e. no data.

3.42 STATE VECTORS FROM DSN NAV

3.42.1 HEADER INFORMATION

Element Type: ODE

Format provided by: CSC

Data provided by: JPL

Data Routinely Accessed by: OFLS (ISS)

Format Structure: ASCII, Sequential

Delivery Media: NASCOM block

3.42.2 CONTENT

This element contains the state vectors of the AXAF spacecraft orbit generated by the DSN Navigation facility at JPL.

3.42.3 FORMAT DESCRIPTION

FILE ORGANIZATION: Sequential

FILE SIZE (ESTIMATED): 732

RECORD LAYOUT:

Record Identifier: DSN State Vector Record

Record Format: Fixed

Record Length: 122 (Bytes)

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_sv_spcsrc	Defines the source of the data	Integer		1
odb_sv_trnsf	Defines the transfer type	Integer		1
odb_sv_icoord	Defines the coordinate system	Integer		1
odb_sv_isic	Defines the support identification code or DSN spacecraft number	Integer		4
odb_sv_ibodnum	Defines the body number or vehicle identifier	Integer		2
odb_sv_vectrcnt	Defines the vector transfer count; Increments by one for each state vector transmitted from the DSN	Integer		3
odb_sv_vrepoch	Define the year of the state vector epoch	Integer		1
odb_sv_doy	Defines the day of year of the state vector epoch (1 January = 001)	Integer		3
odb_sv_epoctime	Defines the state vector epoch time (HHMMSSSSS)	Integer		9
odb_sv_xcomp	Defines the x-component of the spacecraft position (meters)	Integer		13
odb_sv_ycomp	Defines the y-component of the spacecraft position (meters)	Integer		13
odb_sv_zcomp	Defines the z-component of the spacecraft position (meters)	Integer		13

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_sv_xvel	Defines the x-component of the spacecraft velocity (millimeters/second)	Integer		13
odb_sv_yvel	Defines the y-component of the spacecraft velocity (millimeters/second)	Integer		13
odb_sv_zvel	Defines the z-component of the spacecraft velocity (millimeters/second)	Integer		13
odb_sv_spmass	Defines the mass of the spacecraft (kilograms)	Integer		8
odb_sv_sparea	Defines the average spacecraft cross-sectional area (meters^2)	Integer		5
odb_sv_dgftr	Defines the drag factor	Real		4
odb_sv_srpc	Defines the solar radiation pressure coefficient	Real		4

3.43 TABLES, ACIS

3.43.1 HEADER INFORMATION

Element Type: ODE

Format provided by: ASC

Data provided by: ASC

Data routinely accessed by: OFLS (CM)

Format structure: Binary, sequential

Delivery media: Electronic transfer, from the ASC data system

3.43.2 CONTENT

The ACIS tables are sets of command packets, containing either parameter blocks or software commands, used on-board by the AXAF CCD Imaging Spectrometer (ACIS) back-end processor (BEP) to control ACIS configuration and on-board data processing during ACIS observations and other on-board operations. The parameter blocks are the primary mechanism by which configuration data is passed to the ACIS instrument. They are created and updated by the ASCDS, and are formatted by the OFLS for uplink to the spacecraft as part of a command load. Certain of the parameter blocks are stored for extended periods in the ACIS memory, and others are uploaded when needed. Certain parameter blocks are observation-specific, and are designated for use during an ACIS observation by the SI_MODE parameter in the Observation Request (see Section 3.24).

There are three classes of parameter blocks, defined below.

I. ACIS System Configuration Parameter Block, containing hardware settings used to control the CCD (Charge Coupled Device) clocks, control the analog signal processing, set focal plane and housing temperatures, control CCD power restoration following a radiation-induced shutdown, and define bad pixels and columns in the CCD pixel arrays. This block contains relatively fixed parameters, and is expected to be updated infrequently.

II. ACIS Observation Parameter Blocks, used by the BEP software to control the ACIS observing mode by controlling:

   A. Event data collection, in either Timed Exposure or Continuous Clocking Mode. Separate parameter blocks provided for these two modes control selection of the CCD imaging sections, imaging section exposure timing, CCD readout dynamics, and processing of event data, in each mode.

   B. Selection of CCD imaging section sub-regions from which data is to be processed, within the general observing modes described in (i). Sub-regions are defined as either rectangular arrays ("two-dimensional windows") or as groups of columns ("one-dimensional windows") within a CCD. A separate parameter block, associated with the parameter blocks defined in section 3.2, is provided, which defines the sub-regions on all active CCDs. Standard ACIS Observation Parameter Blocks will be provided, plus special blocks keyed to a specific OR when the science objectives of an observation require. Use of the parameter blocks will be specified from the OR by the “ACIS Mode” parameter.

III. ACIS Housekeeping Telemetry Parameter Block, defining additional data to be provided in the telemetry stream.

A naming convention is used to differentiate between SI configurations for observing runs with and without bias measurements. Configurations for observing runs that differ only in whether or not a bias measurement is performed will have the same base name. An “_B” (underscore, uppercase B) is appended to the configuration data file record names for observing runs preceded by a bias measurement. For example, if TEMODE references a configuration for an observing run without a bias measurement, then TEMODE_B is an identical observing run preceded by a bias measurement.

The ACIS Tables data element consists of 2 files: an ACIS configuration file and the ACIS table data file. The ACIS configuration file references up to 16 command packets or parameter blocks and associated timing information used to configure ACIS and start the observation. The command packets or parameter blocks are issued in the order specified in the referenced data record. The timing information is used to delay a subsequent command packet or parameter block by the number of seconds defined. That is, if command packet, A, specified with a 10 second timing delay, is followed by a second command packet, B, then B is issued 10 seconds after A.

The SI_MODE parameter on the OBS or CAL statement from the Observation Request List or the Engineering Request List defines the name used to reference a record from the ACIS configuration file. The SI_MODE parameter specifies the base name for the ACIS

configuration file record when performing an observation without an included ACIS bias measurement. If the observation specified by the OBS or CAL, includes an ACIS bias measurement, then the string "_B" is appended the value of the SI_MODE parameter and used as the name of the record. As an example, if the SI_MODE parameter on an OBS Statement is "TENOMINL", then to perform an observation without a preceding ACIS bias, the ACIS parameter block TENOMINL is referenced. If the referenced parameter block ID is TENOMINL_B, then the observation will include a bias measurement prior to the observation.

There are three ACIS configuration file records referenced by the OFLS MPS using a fixed name:

STOPSCI is used to define the sequence of parameters blocks and/or command packets required to stop a science run.

ERAD is used to define the sequence of parameter blocks and/or command packets required to configure ACIS on predicted entry into the radiation zone. If no parameter block and/or command packet is defined using the name ERAD, no special commanding will be issued by the OFLS MPS on entry into the radiation zone.

XRAD is used to define the sequence of parameter blocks and/or command packets required to configure ACIS on predicted exit from the radiation zone. If no parameter block and/or command packet is defined using the name XRAD, no special commanding will be issued by the OFLS MPS on exit from the radiation zone.

The ACIS data file provides header information and binary data for each parameter block or command packet. The slot id and the command opcode are used to determine if a parameter block must be included in the command load. If the slot id is not blank, then the parameter block is included in the command load if the name of the parameter block currently loaded in the slot of the type identified by the command opcode is different. A packet word length of 0 causes the currently loaded parameter block name to be updated, but no data is included in the command load.

The initial parameter blocks will be developed by the ASC prior to launch, and provided to MSFC for installation into the ODB by the MOL (TBR by MOL). During on-orbit operations, updates and extensions will be provided by the ASCDS by block replacement or addition.

3.43.3 FORMAT DESCRIPTION

The element consists of 2 flat files. The ACIS configuration file is identified by a file extension of ".cfg" and is made up of a set of records, one record for each ACIS configuration (Table 3-13):

Table 3-13 ACIS Configuration File Format

Name	*Description*	*Field Format*	*Field Length (Bytes)*
Configuration Identifier	This identifies ACIS configuration.	ASCII	10
Packet Count	This identifies the number of additional command packets defined in the following	ASCII	2

Name	*Description*	*Field Format*	*Field Length (Bytes)*
	arrays to be used in configuring the instrument. Allowed values are 1 to 16.		
Packet Identifier List	This identifies the set of additional packets that may be included in the command load as necessary to configure the instrument. Packet identifiers specified in this list must be defined in the data element. Up to 16 packets or parameters blocks may be specified.	ASCII	160
Timing Delay	This identifies the timing delay between a packet and a subsequent packet. These values will be used as the timing delay for the associated command packet or parameter blocks in the Packet Identifier List. That is, the 5th value in the list will be used as the timing delay after the 5th command packet or parameter block in the Packet Identifier List. Allowed values are greater than or equal to zero. (seconds)	ASCII	64

The ACIS data file is identified by the file extension of ".dat" and is made up of a set of records, one for each parameter block or command packet (Table 3-14):

Table 3-14 ACIS Data File Format

Name	*Description*	*Field Format*	*Field Length (Bytes)*
Packet Word Length	This indicates the total number 16-bit words in the command packet. Allowed values are 0 and 6 to 512 bytes, where a value of 0 indicates that the associated packet is not to be loaded to ACIS.	ASCII	3
Packet Identifier	This identifies the command packet.	ASCII	10
Command Opcode	This identifies the "Load <type> Parameter Block" field within the parameter block binary data used to instruct the ACIS science Instrument Software to store the data within the packet. Allowed values are "T" for timed exposure observations, "C" for continuous clocking observations, "D" for DEA housekeeping blocks, "1" for 1-dimension window blocks, "2" for 2-dimension window blocks, or blank if not applicable.	ASCII	1
Slot ID	This identifies the parameter block slot within ACIS to overwrite. Allowed values are 0 – 4, or blank if not applicable	ASCII	1
Spare	Not Used	ASCII	1

Name	*Description*	*Field Format*	*Field Length (Bytes)*
Binary Data	This provides the fully formatted binary data to be loaded in the stored command load. The field is variable length with a length as defined in the Packet Word Length.	Binary	6 to 512

3.44 DELETED

3.45 DELETED

3.46 TELEMETRY DEFINITION TABLES

3.46.1 HEADER INFORMATION

Element Type: TDB

Format Provided By: MOL

Data Provided By: TRW

Data Routinely Accessed By: ONLS (DBCR, CMD Update Application)

Format structure: Relational Database Management System (RDBMS) table format

Delivery Media:

Initially the ASCII text files will be file transferred from a workstation in Redondo Beach to a workstation at the OCC.

3.46.2 CONTENT

The Telemetry Definition Tables are used by the EHS to support the telemetry processing system. The tables provide the description of the telemetry stream and information needed for processing the stream. The information provided in the Telemetry Definition tables includes owners of measurements, definition of measurements, measurement sampling and location information, calibration information, limit sensing and exception monitoring information, Virtual Channel Data Unit (VCDU) definition, and the Time Division Multiplexed (TDM) and TDM format definitions. TRW will provide the Telemetry Definition Tables to be used for supporting the AXAF Project.

3.46.3 FORMAT DESCRIPTION

When the Telemetry Definition Tables are loaded onto the EHS they will be in a RDBMS table format. These relational tables are defined in the MSFC HOSC Telemetry Database Definition (MSFC-DOC-1949 Volume 1). However, TRW will provide the tables in an ASCII text file format which is also defined in the MSFC-DOC-1949 Volume 1. In this format there

will be an ASCII text file for each one of the relational tables. For a detailed definition see the MSFC-DOC-1949 Volume 1.

3.47 HRC MODE

3.47.1 HEADER INFORMATION

Element Type:	ODE
Format Provided By:	ASC
Data Provided By:	ASC
Data Routinely Accessed By:	OFLS (MPS)
Format Structure:	ASCII, Sequential
Delivery Media:	Electronic Transfer

3.47.2 CONTENT

The HRC MODE data element contains the HRC configuration parameters used by the OFLS MPS software to configure the HRC detectors (I and S) for an observation. Each record contains a record identifier and a set of parameters and parameter values. The OFLS MPS uses the SI_MODE parameter on the OR OBS statement or the ER CAL statement to access the record with the same record identifier from the HRC MODE data element. The parameters and the parameter values are written in the detailed operations timeline (DOT) on the HRC OBS statement for command load generation processing by the OFLS CM subsystem.

The expected set of parameters is provided in the format description below. However, the OFLS MPS will copy the string of parameters and values verbatim to the DOT. Checking field names and values will be performed as part of the normal command load generation processing. The parameter names in the HRC MODE data element must match the command field names in the CDB or multipart command database used by the OFLS CM to build the HRC instrument commands for uplink to the spacecraft or parameters used in the HRC command sequence definition to control command processing. This means that parameter names are restricted to 20 characters. The parameter values must match the field value type in the CDB or the multipart command database used by the OFLS CM to build the HRC instrument commands for uplink to the spacecraft or parameters used in the HRC command sequence definitions to control command processing.

3.47.3 FORMAT DESCRIPTION

FILE ORGANIZATION:	Sequential
FILE SIZE (ESTIMATED):	43,000
RECORD LAYOUT:	
Record identifier:	HRC MODE Parameter Record
Record Format:	Variable

Record length (maximum); 430

Record Description:

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
odb_hrc_recid	Record identifier as defined on the SI_MODE parameter for the OR OBS statement or the ER CAL statement.	ASCII		8
odb_hrc_parms	String of parameters and parameter values for controlling an HRC observation. The parameters must be in a format compatible with the command sequence substitutable parameter format: parameter=value, The string will be extracted verbatim and included in the DOT. Expected parameters are: Trigger_Level range 0-255 default 20 (TBR) HRC-I or 20 (TBR) HRC-S; data value sent in serial digital command 2LLDIATH Range_Switch_Level range 0-255 default 90 (TBR) HRC-I or 90 (TBR) HRC-S; data value sent in serial digital command 2RSFAATH Spect_Mode (affects HRC-S only) range Normal/Imaging: value of Normal – use serial digital command 2SPMDASL value of Imaging – use serial digital command 2SPNLASL Antico_Enable range Yes/No – default No; value of No – use serial digital command 2SHLIADI value of Yes – use serial digital command 2SHLIAEN			

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
	Width_Enable range Yes/No – default No; value of No – use serial digital command 2WDTHADI value of Yes – use serial digital command 2WDTHAEN Width_Threshold range 0-60 default 3 (TBR) HRC-I or 3 (TBR) HRC-S; data value sent in serial digital command 2WDTHATH ULD_Enable range Yes/No – default No; value of No – use serial digital command 2ULDIADI value of Yes – use serial digital command 2ULDIAEN Upper_Level_Disc range 0-255 default 255 (TBR) HRC-I or 255 (TBR) HRC-S; data value sent in serial digital command 2ULDIATH Blank_Enable range Center/Edge/None – default None: Value of center – use serial digital commands 2CBLKAEN and 2EBLKADI value of Edge – use serial digital commands 2EBLKAEN and 2CBLKADI value of None – use serial digital command 2CBLKADI and 2EBLKADI Note: simultaneous center and edge blanking must not be allowed			

Name	*Description*	*Field Format*	*Dimension*	*Field Length (Bytes)*
	U_Blank_Hi commandable range: 0-255 HRC-I: range 0-60 default 0 (TBR) HRC-S: range 0-15 default 0 (TBR) data value sent in serial digital command 2CBLUALV V_Blank_Hi commandable range 0-255; HRC-I: range 0-60 default 60 (TBR) HRC-S: range 0-190 default 190 (TBR) data value sent in serial digital command 2CBHVALV V_Blank_Low commandable range 0-255; HRC-I: range 0-60 default 0 (TBR) HRC-S: range 0-190 default 0 (TBR) data value sent in serial digital command 2CBLVALV +Y_Shutter_Position_LSB range 0-1600 default 0; value to be inserted into command sequences that steps +Y shutter a commanded number of steps from “Home” +Y_Shutter_Position_MSB range 0-1600 default 0; value to be inserted into command sequences that steps +Y shutter a commanded number of steps from “Home”			

3.48 ACA ARRAYS

3.48.1 HEADER INFORMATION

Element type:	ODE
Format provided by:	ASC
Data provided by:	ASC
Data routinely accessed by:	OFLS (MPS)
Format structure:	FITS format
Delivery media:	Electronic Transfer

3.48.2 CONTENT

The ACA Arrays data element is delivered as a set of three FITS Binary files containing the ACA error array data. The three files are as follows:

FILENAME.CTI provides the charge transfer inefficiency contribution to the ACA positional errors for each pixel

FILENAME.OFA provides the maximum percent sigma increase in star positional uncertainty from off-axis effects to the ACA positional errors

FILENAME.SDP provides the additive ACA error dependent contribution to the ACA positional errors for each pixel

In each case, the file name can be a maximum of 25 characters excluding the 4 character file type extension. The file extensions of CTI, OFA, and SDP must be specified.

Each file contains a 1024 by 1024 array of values representing the error contribution for each pixel in the ACA CCD. Each file starts with a variable number of 2880 byte header records, the last record starting with the string “END”. The OFLS ignores all data up to and including the header END record.

3.48.3 FORMAT DESCRIPTION

FILE ORGANIZATION: ASCII, FITS format

FILE SIZE (ESTIMATED): 8,4000,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Charge Transfer Inefficiency Blur Factor Header Record

Record Format: Fixed

Record Length: 2880 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_cti_header	Defines a FITS header record. The first three characters of the last FITS header record have the value, END.	ASCII		2880

RECORD LAYOUT:

Record Identifier: Charge Transfer Inefficiency Blur Factor Record

Record Format: Fixed

Record Length: 8388608 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_cti_blur_fact	Defines the charge transfer induced blur factor for each pixel index: 1 = y-pixel (minimum to maximum) , 2 = z-pixel (minimum to maximum)	Real	1024x1024	8388608

FILE ORGANIZATION: ASCII, FITS format

FILE SIZE (ESTIMATED): 8,400,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Maximum Off-axis Error Contribution Header Record

Record Format: Fixed

Record Length: 2880 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_offax_header	Defines a FITS header record. The first three characters of the last FITS header record have the value, END.	ASCII		2880

RECORD LAYOUT:

Record Identifier: Maximum Off-axis Error Contribution Record

Record Format: Fixed

Record Length: 8388608 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_max_offax_sigma	Defines the maximum percent sigma increase in star positional uncertainty from off-axis effect for each pixel; index: 1 = y-pixel (minimum to maximum) , 2 = z-pixel (minimum to maximum)	Real	1024 x 1024	8388608

FILE ORGANIZATION: ASCII, FITS format

FILE SIZE (ESTIMATED): 500,000 (Bytes)

RECORD LAYOUT:

Record Identifier: Additive Error Contribution Header Record

Record Format: Fixed

Record Length: 2880 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_add_err_header	Defines a FITS header record. The first three characters of the last FITS header record have the value, END.	ASCII		2880

RECORD LAYOUT:

Record Identifier: Additive Error Contribution Record

Record Format: Fixed

Record Length: 8388608 (Bytes)

Record Description:

Name	Description	Field Format	Dimension	Field Length (Bytes)
odb_aca_err_array	Defines the additive ACA error array; index: 1 = y-pixel (minimum to maximum) , 2 = z-pixel (minimum to maximum)	Real	1024 x 1024	8388608

4. ODB MAINTENANCE, CONTROL, USE, AND VERIFICATION Instructions

4.1 ODB CONFIGURATION CONTROL AND MAINTENANCE

The procedures for configuration control and maintenance of the ODB are specified in the Advanced X-ray Astrophysics Facility - Imaging (AXAF) Operations Database (ODB) Management Plan Document (AMO-1140).

4.2 ACCESSING ODB ELEMENTS

The Telemetry Definition Tables can be accessed through the Telemetry Database (TDB) Application and the Command Database Definition Tables can be accessed through the Command Database (CDB) Application.

User's Guides for the EHS Launch pad and each of the EHS applications (both ONLS and OFLS) provide the user with information needed to access ODB elements.

APPENDIX A Data Elements accessing overview

This appendix provides an overview of how each of the data elements can be accessed using various software system(s)/tool(s) available at the ground system. The table consists of 10 columns with the following definitions:

Column Name	Description
Type	Data element type (CDB, ODE, TDB) CDB: Command Database ODE: Off-Line Data Element TDB: Telemetry Database
No	Data element number, corresponds to 2nd level section numbers for section 3
ODB Element	Data element name, detail definition is described in section 3
Format Source (Org.)	Format provider organization (ASC, CSC, JPL, MOL, TRW) ASC: AXAF Science Center (acting Science Operations Team) CSC: OFLS developer JPL: DSN related data provider MOL: ONLS developer TRW: Spacecraft operations(acting Flight Operations Team)
Data Originator	Data provider organization. Organization list is the same as the format source (org.) field plus the following FOT: Flight Operations Team SMF: Software Maintenance Facility SOT: Science Operations Team
Routinely Accessed By (Org.)	Organization(s) routinely accessing the element. This is either FOT or SOT or both.
Routinely Accessed By (S/W)	S/W system(s) routinely accessing the element. This column defines the S/W system accessing the data element for processing. ASC Tool: S/W tools created and maintained by the SOT for editing and viewing data elements. These are usually data elements in Binary or ASCII file format where no GS S/W tools other than FRAMEMAKER is available. CDB APP: Command database application FOT Tool: S/W tools created and maintained by the FOT for editing and viewing data elements. These are usually data elements in Binary or ASCII file format where no GS S/W tools other than FRAMEMAKER is available. FRAMEMAKER: Document editor available as an ODE tool for editing and viewing ASCII format data elements N/A: Not applicable, no software tool(s) available OFLS: Off-Line System software items - AD&SC: Attitude Determination and Spacecraft Control - CM: Command Management - ISS: Interface & Support Software - MPS: Mission Planning and Scheduling - SS&EA: Spacecraft Support & Engineering Analysis ONLS: On-Line System software - DBCR: Database Change Request software - CM Update APP: Command Update Application - Mission Comp: Mission computation

Column Name	Description
	- ODE: On-Line System ODE Tool, generally used for storing and retrieving data elements to and from the database server. - WCP/CCP: Workstations command processor/Central command processor TDB APP: Telemetry database application
ODB Input S/W (Tools)	S/W system(s)/tool(s) available for creating and storing the data element into database server.
ODB Output S/W (Tools)	S/W system(s)/tool(s) available for viewing the data element
Maintenance S/W (Tools)	S/W system(s)/tool(s) available for editing or maintaining the data element

Table A-1 ODB Data Elements and Accessing S/W

TYPE	NO	ODB ELEMENT	FMT SOURCE (ORG.)	DATA ORIGINATOR	ROUTINELY ACCESSED BY (S/W)	ROUTINELY ACCESSED BY (ORG.)	ODB INPUT S/W (TOOLS)	ODB OUTPUT S/W (TOOLS)	MAINTENANCE S/W (TOOLS)
ODE	1	Not Used							
ODE	2	CHARACTERISTICS	CSC	FOT, SOT, CSC	OFLS (ALL)	FOT, SOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
CDB	3	COMMAND DEFINITION TABLES	MOL	FOT	ONLS (DBCR, CMD UPDATE APP.), OFLS (CM)	FOT, SOT	CDB APP.	CDB APP.	CDB APP.
ODE	4	COMMAND DEFINITION TABLES, MULTI-PART	CSC	FOT	OFLS (CM)	FOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
ODE	5	COMMAND LOAD	CSC	OFLS (CM)	ONLS, OFLS (CM)	FOT	OFLS (CM)	ONLS (WCP/CCP)	OFLS (CM)[1]
ODE	6	COMMAND LOAD IMAGE	CSC	OFLS (CM)	ONLS, OFLS (CM)	FOT	OFLS (CM)	ONLS (MISSION COMP.)	OFLS (CM)[5]
ODE	7	COMMAND SEQUENCE DEFINITIONS	CSC	FOT, SOT	OFLS (CM)	FOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
ODE	8	CONFIGURATION REFERENCE	CSC	FOT	OFLS (CM)	FOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
ODE	9	CONFIGURATION SNAPSHOT	CSC	OFLS (CM)	ONLS (MISSION COMP), OFLS (CM)	FOT	OFLS (CM)	ONLS	OFLS (CM)[5]
ODE	10	CONSTRAINTS	CSC	FOT, SOT, CSC	OFLS (ALL)	FOT, SOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
ODE	11	DSN APPROVED SCHEDULES	JPL	JPL	OFLS (MPS, CM)	FOT, SOT	ONLS (ODE)[2]	FRAMEMAKER	N/A
ODE	12	DSN SCHEDULE REQUESTS	JPL	OFLS (MPS)	OFLS (MPS)	FOT	OFLS (MPS)	ONLS (ODE)[6]	OFLS (MPS)[5]
ODE	13	ENGINEERING REQUEST	CSC	FOT, SOT	OFLS (MPS)	FOT, SOT	ONLS (ODE),	OFLS (MPS),	OFLS (MPS),

1 re-run the software for new update, no editing allowed

2 utilizing ONLS (ODE) store/retrieve for file transfers between database server and other sources (ex. SMF, PC with Internet access)

Table A-1 ODB Data Elements and Accessing S/W

TYPE	NO	ODB ELEMENT	FMT SOURCE (ORG.)	DATA ORIGINATOR	ROUTINELY ACCESSED BY (S/W)	ROUTINELY ACCESSED BY (ORG.)	ODB INPUT S/W (TOOLS)	ODB OUTPUT S/W (TOOLS)	MAINTENANCE S/W (TOOLS)
							OFLS (MPS)	FRAMEMAKER	FRAMEMAKER
ODE	14	EIOEMERIS, DEFINITIVE	CSC	OFLS (ISS)	OFLS (ALL)	FOT, SOT	OFLS (ISS)	OFLS (ISS) to create the report, FRAMEMAKER for viewing	OFLS (ISS)[5]
ODE	15	EIOEMERIS, PREDICTIVE	CSC	OFLS (ISS)	OFLS (ALL)	FOT, SOT	OFLS (ISS)	OFLS (ISS) to create the report, FRAMEMAKER for viewing	OFLS (ISS)[5]
ODE	16	MCILWAIN PARAMETERS	CSC	OFLS (ISS)	ASC TOOL	FOT, SOT	OFLS (ISS)	N/A	OFLS (ISS)[5]
ODE	17	MEMORY IMAGE, AC	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	18	MEMORY IMAGE, CPE	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	19	MEMORY IMAGE, CTU EEPROM	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	20	MEMORY IMAGE, I-EPHIN	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	21	MEMORY IMAGE, IU EEPROM	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	22	MEMORY IMAGE, OBC	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	23	MEMORY IMAGE, SIM	TRW	SMF	ONLS (MISSION COMP)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	24	OBSERVATION REQUEST	CSC	SOT	OFLS (MPS)	FOT, SOT	ONLS (ODE), OFLS (MPS)	OFLS (MPS), FRAMEMAKER	OFLS (MPS), FRAMEMAKER

Table A-1 ODB Data Elements and Accessing S/W

TYPE	NO	ODB ELEMENT	FMT SOURCE (ORG.)	DATA ORIGINATOR	ROUTINELY ACCESSED BY (S/W)	ROUTINELY ACCESSED BY (ORG.)	ODB INPUT S/W (TOOLS)	ODB OUTPUT S/W (TOOLS)	MAINTENANCE S/W (TOOLS)
ODE	25	ORBIT EVENTS, DEFINITIVE	CSC	OFLS (ISS)	OFLS (ALL except AD&SC)	FOT, SOT	OFLS (ISS)	OFLS (ISS) to create the report	OFLS (ISS)[5]
ODE	26	ORBIT EVENTS, PREDICTIVE	CSC	OFLS (ISS)	OFLS (ALL except AD&SC)	FOT, SOT	OFLS (ISS)	OFLS (ISS) to create the report	OFLS (ISS)[5]
ODE	27	RADIATION ZONE DEFINITIONS	CSC (NSSDC)	FOT (NSSDC)	OFLS (ISS)	FOT	ONLS (ODE)[6]	N/A	N/A
ODE	28	RELATIVE TIME SEQUENCE	CSC	FOT, SOT	OFLS (CM)	FOT	ONLS (ODE)	FRAMEMAKER	FOT TOOL
ODE	29	SCHEDULED OR/ER DATA	CSC	OFLS (MPS)	ASC TOOL	SOT	OFLS (MPS)	FRAMEMAKER	OFLS (MPS)[5]
ODE	30	SENSOR CALIBRATION DATA	CSC	OFLS (AD&SC)	OFLS (AD&SC)	FOT	OFLS (AD&SC)	N/A	OFLS (AD&SC)[5]
ODE	31	SOFTWARE UPDATES, AC	TRW	SMF	OFLS (CM)	FOT, SOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	32	SOFTWARE UPDATES, ACIS	ASC	ASC	OFLS (CM)	FOT, SOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	33	SOFTWARE UPDATES, CPE	TRW	SMF	OFLS (CM)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	34	SOFTWARE UPDATES, CTU EEPROM	TRW	SMF	OFLS (CM)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	35	SOFTWARE UPDATES, I-EPHIN	TRW	SMF	OFLS (CM)	FOT, SOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	36	SOFTWARE UPDATES, IU EEPROM	TRW	SMF	OFLS (CM)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	37	SOFTWARE UPDATES, OBC	TRW	SMF	OFLS (CM)	FOT	ONLS (ODE)[6]	FOT TOOL	SMF
ODE	38	SOFTWARE UPDATES, SIM	TRW	SMF	OFLS (CM)	FOT, SOT	ONLS (ODE)[6]	FOT TOOL	SMF

Table A-1 ODB Data Elements and Accessing S/W

TYPE	NO	ODB ELEMENT	FMT SOURCE (ORG.)	DATA ORIGINATOR	ROUTINELY ACCESSED BY (S/W)	ROUTINELY ACCESSED BY (ORG.)	ODB INPUT S/W (TOOLS)	ODB OUTPUT S/W (TOOLS)	MAINTENANCE S/W (TOOLS)
ODE	39	SOLAR, LUNAR, PLANETARY DATA	CSC	JPL	OFLS (ALL except CM)	FOT, SOT	ONLS (ODE)[6]	N/A	N/A
ODE	40	SPACECRAFT CLOCK CORRELATION	CSC	OFLS (SS&EA)	OFLS (ALL)	FOT, SOT	OFLS (SS&EA)	OFLS (SS&EA)	OFLS (SS&EA)[5]
ODE	41	STAR CATALOG	ASC, CSC	SOT	OFLS (MPS, AD&SC, SS&EA)	FOT, SOT	ONLS (ODE)	N/A	ASC
ODE	42	STATE VECTORS FROM DSN NAV	CSC	JPL	OFLS (ISS)	FOT	ONLS (MISSION COMP.)	FRAMEMAKER	N/A
ODE	43	TABLES, ACIS	ASC	SOT	OFLS (CM)	FOT, SOT	ASC TOOL	ASC TOOL	ASC TOOL
ODE	44	Deleted							
ODE	45	Deleted							
TDB	46	TELEMETRY DEFINITION TABLES	MOL	FOT	ONLS (DBCR, CMD UPDATE APP.)	FOT, SOT	TDB APP.	TDB APP.	TDB APP.

FootNotes:

NOTE. 4 RE-RUN THE SOFTWARE FOR NEW UPDATES, NO EDITING ALLOWED.

NOTE 5. UTILIZING ONLS (ODE) STORE/RETRIEVE FOR FILE TRANSFERS BETWEEN DATABASE SERVER AND OTHER SOURCES (EX. SMF, PC WITH INTERNET ACCESS)